



TUD-KOMA-Script v2.05 basierend auf KOMA-Script

EIN $\text{L}^{\text{A}}\text{T}_{\text{E}}\text{X}$ 2 $_{\epsilon}$ -BUNDLE FÜR DOKUMENTE IM NEUEN CORPORATE DESIGN DER TECHNISCHEN UNIVERSITÄT DRESDEN

Benutzerhandbuch*

Falk Hanisch[†]

2. Juli 2015

*Druckversion

[†]tudscr@gmail.com

VORWORT

Die im Folgenden beschriebenen Klassen und Pakete wurden für das Erstellen von \LaTeX -Dokumenten im Corporate Design der Technischen Universität Dresden entwickelt.¹ Diese basieren auf den gerade im deutschsprachigen Raum häufig verwendeten KOMA-Script-Klassen, welche eine Vielzahl von Einstellmöglichkeiten bieten, die weit über die Möglichkeiten der \LaTeX -Standardklassen hinausgehen. Zusätzlich bietet das hier dokumentierte TUD-KOMA-Script-Bundle weitere, insbesondere das Dokumentlayout betreffende Auswahlmöglichkeiten.

Es sei angemerkt, dass die hier beschriebenen Klassen eine Abweichung vom Corporate Design der Technischen Universität Dresden zulassen, da dieses gerade unter typografischen Gesichtspunkten durchaus als diskussionswürdig zu erachten ist. Prinzipiell ist es mit den entsprechenden Einstellungen möglich, auf das standardmäßige Layout der KOMA-Script-Klassen zurückzuschalten. Ohne die gezielte Verwendung dieser Optionen durch den Anwender werden per Voreinstellung alle Vorgaben des Corporate Designs umgesetzt.

In diesem Handbuch werden dem Anwender Hinweise für eine einfache Installation der benötigten Schriften des Corporate Designs sowie eine Beschreibung der zusätzlich zu den KOMA-Script-Klassen nutzbaren Optionen und Befehle gegeben. Dabei werden Grundkenntnisse in der Verwendung von \LaTeX 2_ε vorausgesetzt. Sollten diese nicht vorhanden sein, wird das Lesen der \LaTeX 2_ε-Kurzbeschreibung [l2kurz.pdf](#)² dringend empfohlen. Für den stärker vertieften Einstieg in \LaTeX gibt es eine [Workshop-Reihe](#)³ von Fadi Semmo. Außerdem stellt Nicola L. C. Talbot sehr ausführliche Tutorials für [\$\text{\LaTeX}\$ -Novizen](#)⁴ und [Dissertationen](#)⁵ zur freien Verfügung. Außerdem werden in [Teil II](#) dieses Handbuchs Minimalbeispiele sowie etwas ausführlichere Tutorials angeboten.

Des Weiteren sollte *jeder* Anwender das \LaTeX 2_ε-Sündenregister [l2tabu.pdf](#)⁶ kennen, um typische Fehler zu vermeiden. Antworten auf häufig gestellte Fragen liefert [DANTE" FAQ](#)⁷. Falls der Nutzer unerfahren bei der Verwendung von KOMA-Script sein sollte, so ist ein Blick in das [dazugehörige Handbuch](#)⁸ sehr zu empfehlen, wenn nicht sogar unumgänglich.

Der aktuelle Stand der Klassen und Pakete aus dem TUD-KOMA-Script-Bundle in der Version v2.05 wurde nach bestem Wissen und Gewissen auf Herz und Nieren getestet. Dennoch kann nicht für das Ausbleiben von Fehlern garantiert werden. Beim Auftreten eines Problems sollte dieses genauso wie Inkompatibilitäten mit anderen Paketen im Forum unter

<http://latex.wcms-file3.tu-dresden.de/phpBB3/>

gemeldet werden. Für eine schnelle und erfolgreiche Fehlersuche sollte bitte ein [Minimalbeispiel](#)⁹ bereitgestellt werden. Auf Anfragen ohne dieses werde ich gegebenenfalls verspätet oder gar nicht reagieren. Ebenso sind im genannten Forum auch *Fragen*, *Kritik* und *Verbesserungsvorschläge* – sowohl das Bundle selbst als auch die Dokumentation betreffend – gerne gesehen. Da dieses Bundle in meiner Freizeit entstanden ist und auch gepflegt wird, bitte ich um Nachsicht, falls ich nicht sofort antworte und/oder eine Fehlerkorrektur vornehmen kann.

Falk Hanisch

Dresden, 2. Juli 2015

¹<http://tu-dresden.de/cd> http://tu-dresden.de/service/publizieren/cd/6_handbuch/index.html

²<http://mirrors.ctan.org/info/lshort/german/l2kurz.pdf>

³<http://www.fadi-semmo.de/latex/workshop/>

⁴<http://www.dickimaw-books.com/latex/novices/>

⁵<http://www.dickimaw-books.com/latex/thesis/>

⁶<http://mirrors.ctan.org/info/l2tabu/german/l2tabu.pdf>

⁷<http://projekte.dante.de/DanteFAQ/WebHome>

⁸<http://mirrors.ctan.org/macros/latex/contrib/koma-script/doc/scrguide.pdf>

⁹<http://www.komascript.de/minimalbeispiel>

INHALTSVERZEICHNIS

Vorwort	2
1 Einleitung	8
1.1 Zur Verwendung dieses Handbuchs	9
1.2 Installation des TUD-KOMA-Script-Bundles	10
1.2.1 Installation der PostScript-Schriften unter Windows	12
1.2.2 Installation der PostScript-Schriften unter Linux und OS X	12
1.2.3 Probleme bei der Installation der PostScript-Schriften	12
1.3 Schnelleinstieg	13
I Das TUD-KOMA-Script-Bundle	14
2 Die Klassen tudscrbook, tudscrreprt und tudscrartcl	15
2.1 Die Schriften des Corporate Designs	16
2.1.1 Schriften für den Textsatz	17
2.1.2 Schriften für den Mathematiksatz	20
2.1.3 Die Schriften des Corporate Designs im OpenType-Format	21
2.2 Das Layout des Corporate Designs	21
2.2.1 Die Gestalt von Titel, Umschlagseite, Teilen und Kapiteln	22
2.2.2 Seiten im Stil des Corporate Designs	24
2.2.3 Der Titel und die Umschlagseite	29
2.2.4 Die Teileseite	36
2.2.5 Die Kapitelseite	36
2.2.6 Vakatsseiten	37
2.2.7 Der Satzspiegel	38
2.2.8 Verwendung von Schriftelementen	40
2.2.9 Die Farben des Corporate Designs	40
2.3 Zusätzliche Optionen und Erweiterungen	40
2.3.1 Zusammenfassung/Kurzfassung	40
2.3.2 Selbstständigkeitserklärung und Sperrvermerk	43
2.3.3 Fußnoten in Überschriften	48
2.3.4 Lesezeichen	48
2.4 Sprachabhängige Bezeichner	48
2.5 Kompatibilitätseinstellungen zu früheren Versionen	52
3 Die Klasse tudscrposter	54
4 Zusätzliche Pakete im TUD-KOMA-Script-Bundle	56
4.1 Das Paket tudscrsupervisor – Studentische Betreuung	57
4.1.1 Aufgabenstellung für eine wissenschaftliche Arbeit	57
4.1.2 Gutachten für wissenschaftliche Arbeiten	58
4.1.3 Aushang	59
4.1.4 Zusätzliche sprachabhängige Bezeichner	60
4.2 Das Paket tudscrcolor – Farben im Corporate Design	63
4.2.1 Generelle Farbdefinitionen	63
4.2.2 Zusätzliche Farbdefinitionen	65
4.2.3 Umstellung des Farbmodells	65

4.3	Das Paket tudscrfonts – Schriften im Corporate Design	65
4.4	Das Paket mathswap	66
4.5	Das Paket twocolfix	67
II	Ergänzungen und Hinweise	68
5	Minimalbeispiele und Tutorials	69
5.1	Dokument	70
5.2	Dissertation	75
5.3	Abschlussarbeit (kollaborativ)	78
5.4	Aufgabenstellung (kollaborativ)	80
5.5	Gutachten	82
5.6	Aushang	84
5.7	Vorlage für eine wissenschaftlichen Arbeit	86
5.8	Ein Beitrag zum mathematischen Satz in $\text{\LaTeX} 2_{\epsilon}$	86
5.9	Änderung der Trennzeichen im Mathematikmodus	86
6	Unerlässliche und beachtenswerte Pakete	87
6.1	Von den neuen Hauptklassen benötigte Pakete	88
6.1.1	Erforderliche Pakete bei der Schriftinstallation	88
6.1.2	Notwendige Pakete für die Verwendung der Hauptklassen	88
6.2	Durch TUD-KOMA-Script direkt unterstützte Pakete	89
6.3	Empfehlenswerte Pakete	90
6.3.1	Pakete zur Verwendung in jedem Dokument	90
6.3.2	Pakete zur situativen Verwendung	91
7	Praktische Tipps & Tricks	102
7.1	\LaTeX -Editoren	103
7.2	Literaturverwaltung in \LaTeX	103
7.3	Worttrennungen in deutschsprachigen Texten	104
7.4	Lokale Änderungen von Befehlen und Einstellungen	105
7.5	Bezeichnung der Gliederungsebenen durch hyperref	105
7.6	URL-Umbrüche im Literaturverzeichnis mit biblatex	105
7.7	Zeilenabstände in Überschriften	106
7.8	Warnung wegen zu geringer Höhe der Kopf-/Fußzeile	106
7.9	Einrückung von Tabellenspalten verhindern	106
7.10	Unterdrückung des Einzuges eines Absatzes	107
7.11	Unterbinden des Zurücksetzens von Fußnoten	107
7.12	Setzen von Einheiten mit siunitx	107
7.13	Warnung beim Erzeugen des Inhaltsverzeichnisses	107
7.14	Leer- und Satzzeichen nach \LaTeX -Befehlen	108
7.15	Das Setzen von Auslassungspunkten	108
7.16	Finden von unbekannten \LaTeX -Symbolen	109
7.17	Änderung des Papierformates	109
7.18	Vermeiden des Skalierens einer PDF-Datei beim Druck	109
7.19	Warnung bei der Schriftgrößenwahl	110
7.20	Platzierung von Gleitobjekten	110

7.21 Automatisiertes Einbinden von <i>Inkscape</i> -Grafiken	112
7.22 Fehlermeldung: ! No room for a new \write	112
7.23 Fehlermeldung beim Laden eines Paketes mit Optionen	113
7.24 Probleme bei der Verwendung von auto-pst-pdf	113
III Anhang	115
A Weiterführende Installationshinweise	116
A.1 Lokale Deinstallation des TUD-KOMA-Script-Bundles	118
A.2 Lokale Installation des TUD-KOMA-Script-Bundles	119
A.2.1 Lokale Installation von TUD-KOMA-Script unter Windows	119
A.2.2 Lokale Installation von TUD-KOMA-Script unter Linux und OS X	119
A.3 Lokales Update des TUD-KOMA-Script-Bundles	120
A.3.1 Update des TUD-KOMA-Script-Bundles ab Version v2.02	120
A.3.2 Update des TUD-KOMA-Script-Bundles ab Version v2.00	120
A.3.3 Update des TUD-KOMA-Script-Bundles von Version v1.0	120
A.4 Installationshinweise für portable Installationen	121
B Obsolete sowie vollständig entfernte Optionen und Befehle	122
B.1 Veraltete Optionen und Befehle in TUD-KOMA-Script	123
B.1.1 Änderungen für TUD-KOMA-Script v2.00	123
B.1.2 Änderungen für TUD-KOMA-Script v2.02	125
B.1.3 Änderungen für TUD-KOMA-Script v2.03	126
B.1.4 Änderungen für TUD-KOMA-Script v2.04	126
B.2 Das Paket tudscrcomp – Umstieg von anderen Klassen	127
B.2.1 Optionen und Befehle aus tudbook & Co.	127
C Identifikation von TUD-KOMA-Script	129
Index	131
Allgemeiner Index	132
Klassen- und Paketoptionen	135
Umgebungen, Befehle und Seitenstile	136
Sprachabhängige Bezeichner	140
Schriftelemente	141
Farben	142
Klassen, Pakete und Dateien	143
Änderungsliste	146

KAPITEL 1

EINLEITUNG

Zur fehlerfreien Verwendung der TUD-KOMA-Script-Klassen der Version v2.05 werden sowohl die KOMA-Script-Klassen v3.17 oder später als auch die beiden Hausschriften des Corporate Designs **Univers** und **DIN BOLD** zwingend benötigt. Außerdem müssen weitere Pakete durch die genutzte L^AT_EX-Distribution bereitgestellt werden.

Bei den aktuellen Distributionen T_EX Live 2015, MacT_EX 2015 und MiK_TE_X 2.9 ist das mit großer Sicherheit kein Problem. Nutzen Sie jedoch eine ältere Distribution, könnte dies zu Problemen führen. Dann sollte bestenfalls eine der aktuellen Distributionen installiert werden. Ist dies nicht möglich, müssen die unter [Unterkapitel 6.1](#) aufgeführten Pakete sowie TUD-KOMA-Script ([Unterkapitel A.2](#)) in der jeweils benötigten Version lokal installiert werden.

Das Vorlagenpaket von Klaus Bergmann ist für die Verwendung nicht notwendig. Allerdings beinhaltet dieses weitere Klassen zum Erstellen von Folien und Briefen.¹⁰ Das TUD-KOMA-Script-Bundle ist hauptsächlich für das Erstellen wissenschaftlicher Texte und Arbeiten gedacht und soll die ursprünglichen Vorlagen *momentan* nicht ersetzen sondern vielmehr ergänzen.

Eine Umsetzung des Corporate Designs für die **beamer**-Klasse sowie für Briefe und Geschäftsschreiben auf Basis der KOMA-Script-Brief-Klasse **scr_ltr2** ist bis jetzt leider noch nicht entstanden, soll jedoch langfristig bereitgestellt werden. Allerdings existieren bereits im Bundle **tudmathposter** für die **beamer**-Klasse mehrere Stile. Dieses Bundle ist sowohl bei [GitHub](#)¹¹ als auch auf der [L^AT_EX-Seite der Technischen Universität Dresden](#)¹² zu finden.

1.1 ZUR VERWENDUNG DIESES HANDBUCHS

Sämtliche neu definierten Optionen, Umgebungen und Befehle der TUD-KOMA-Script-Klassen und TUD-KOMA-Script-Pakete werden im Handbuch aufgeführt und beschrieben. Am Ende des Dokumentes befinden sich mehrere Indizes, die das Nachschlagen oder Auffinden von bisher unbekannten Befehlen oder Optionen erleichtern sollen.

Die im Folgenden beschriebenen Optionen können – wie ein Großteil aller Einstellungen für KOMA-Script – in der Syntax des **keyval**-Paketes als Schlüssel-Wert-Paare bei der Wahl der Dokumentklasse angegeben werden: `\documentclass[<Schlüssel>=<Wert>]{<Klasse>}`.

Des Weiteren eröffnen die KOMA-Script-Klassen die Möglichkeit der späten Optionenauswahl. Damit können Optionen nicht nur direkt beim Laden als sogenannte Klassenoptionen angegeben werden, sondern lassen sich auch noch innerhalb des Dokumentes nach dem Laden der Klasse ändern. Die KOMA-Script-Klassen sehen hierfür zwei Befehle vor. Mit `\KOMAOPTIONS{<Optionenliste>}` kann man beliebig vielen Schlüsseln jeweils genau einen Wert zuweisen, `\KOMAOPTION{<Option>}{<Werteliste>}` erlaubt das gleichzeitige Setzen mehrerer Werte für genau einen Schlüssel. Äquivalent dazu werden für die *zusätzlichen* Optionen der TUD-KOMA-Script-Klassen mit die Befehle `\TUOptions{<Optionenliste>}` und `\TUOption{<Option>}{<Werteliste>}` definiert. Damit kann das Verhalten von Optionen im Dokument – innerhalb einer Gruppe auch lokal – geändert werden.

Die Voreinstellung einer jeden Option wird durch `preset: <Wert>` bei deren Beschreibung angeführt. Einige dieser Standardwerte sind nicht immer gleich sondern werden in Abhängigkeit der genutzten Benutzereinstellungen und Optionen gesetzt. Diese bedingten Voreinstellungen werden durch `preset: <Wert> | <Bedingung>: <bedingter Wert>` angegeben. Jedem Schlüssel wird normalerweise durch den Benutzer ein gewünschter, gültiger Wert zugewiesen. Wird ein Schlüssel jedoch ohne Wertzuweisung genutzt, so wird – falls vorhanden – ein vordefinierter

¹⁰`tudbook`, `tudbeamer`, `tudletter`, `tudfax`, `tudhaus`, `tudform`

¹¹<https://github.com/tud-cd/tud-cd>

¹²http://tu-dresden.de/service/publizieren/cd/4_latex

Säumniswert gesetzt, welcher in der Beschreibung der einzelnen Optionen durch die *kursive* Schreibweise gekennzeichnet ist. In den meisten Fällen ist der Säumniswert eines Schlüssels `true`, er entspricht folglich der Angabe `<Schlüssel>=true`. Mit der expliziten Wertzuweisung eines Schlüssels durch den Benutzer werden immer sowohl normale als auch bedingte Voreinstellungen überschrieben. Die neben den Optionen neu eingeführten Befehle und Umgebungen der Klassen werden im gleichen Stil erläutert.

1.2 INSTALLATION DES TUD-KOMA-SCRIPT-BUNDLES

v2.01
v2.02
v2.04

Das TUD-KOMA-Script-Bundle ist seit der Version v2.01 – aufgrund lizenzrechtlicher Bedingungen *ohne* die geschützten Schriften **Univers** und **DIN BOLD** – im **Comprehensive TeX Archive Network (CTAN)**¹³ zu finden und kann dadurch die aktuellen L^AT_EX-Distributionen wie **T_EX Live 2015**, **MacT_EX 2015** oder auch **MiK_TE_X 2.9** genutzt werden. Es besteht momentan aus den drei Hauptklassen **tudscrbook**, **tudscrreprt** sowie **tudscrartcl** – dokumentiert in **Kapitel 2** – und sowohl mit den TUD-KOMA-Script-Klassen assoziierten als auch eigenständigen Paketen, welche in **Kapitel 4** beschrieben sind.

Zur problemlosen Verwendung des TUD-KOMA-Script-Bundles ist – neben KOMA-Script in der Version v3.17 und den in **Unterkapitel 6.1** aufgeführten L^AT_EX-Paketen – lediglich eine Installation der PostScript-Schriften des Corporate Designs der Technischen Universität Dresden notwendig. Diese müssen über das Universitätsmarketing auf **Anfrage**¹⁴ mit dem Hinweis auf die Verwendung von L^AT_EX bestellt werden. Sobald Sie die notwendigen Archive **Univers_PS.zip** und **DIN_Bd_PS.zip** erhalten haben, können die Schriften für Windows (**Abschnitt 1.2.1**) beziehungsweise unixoide Betriebssysteme (**Abschnitt 1.2.2**) installiert werden. Die benötigten Skripte werden als **Release**¹⁵ im **GitHub-Repository tudscr** bereitgestellt.¹⁶

Achtung!

Die Skripte erzeugen für die Schriftinstallation temporär einige Verzeichnisse und Dateien, weshalb diese nur fehlerfrei auf einem Laufwerk ausgeführt werden können, auf dem der Anwender selbst Schreibrechte besitzt.

ANMERKUNG ZU WINDOWS

Sollte Windows genutzt werden und noch keine L^AT_EX-Distribution auf ihrem System installiert sein, so rate ich persönlich zur Verwendung von **T_EX Live** statt **MiK_TE_X**. Der Vorteil dieser Distribution liegt zum einen in der Wartung durch mehrere Autoren sowie der früheren Verfügbarkeit aller Updates über CTAN. Zum anderen liefert **T_EX Live** zusätzlich zu L^AT_EX_{2_ε} einen PERL-Interpreter sowie GHOSTSCRIPT, wodurch die Ad-hoc-Verwendung einiger Pakete wie beispielsweise **glossaries** vereinfacht beziehungsweise verbessert wird.

ANMERKUNG ZU LINUX UND OS X

Die Installation von **T_EX Live** beziehungsweise **MacT_EX** sollte direkt über die im Internet angebotenen Installationspakete (<https://tug.org/texlive/> respektive <https://tug.org/mactex/>) und nicht über `apt-get install` erfolgen. Die Installation der Schriften des Corporate Designs *muss zwingend* über das Terminal ausgeführt werden. Nach dem Entpacken eines Release-Archivs sollte im passenden Pfad (beispielsweise `cd ~/Downloads/<Unterordner>`) das Skript aus dem Terminal mit `bash <Skript>.sh` direkt ausgeführt werden. Mehr dazu in **Abschnitt 1.2.2**.

¹³<http://www.ctan.org/>

¹⁴http://tu-dresden.de/service/publizieren/cd/1_basiselemente/03_hausschrift/schriftbestellung.html

¹⁵<https://github.com/tud-cd/tudscr/releases/tag/fonts>

¹⁶Die Verwendung von installierten Systemschriften im Open-Type-Format mit dem Paket **fontspec** für LuaL^AT_EX oder X₃L^AT_EX wird mittlerweile unterstützt, mehr dazu in **Abschnitt 2.1.3**.

ANMERKUNG ZU M_IK_TE_X

Vor der Installation der Schriften für TUD-KOMA-Script sollte unbedingt ein Update von M_IK_TE_X durchgeführt werden. Andernfalls wird es mit großer Sicherheit zu Problemen kommen. Außerdem ist es sehr ratsam, die Installation von M_IK_TE_X in der Mehrbenutzervariante mit Administratorrechten durchzuführen, da die Einzelbenutzervariante relativ unregelmäßig und nicht immer nachvollziehbar zu Problemen führen kann.

Es kann vorkommen, dass die für den Schriftinstallationsprozess notwendigen Pakete **fontinst**, **fontware** sowie **cmbright**, **hfbright**, **cm-super**, **lmodern** und **iwona** noch nicht installiert sind. Ist die automatische Nachinstallation fehlender Pakete aktiviert, so reicht es im Normalfall das Installationsskript zu starten. Andernfalls müssen diese Pakete manuell durch den Benutzer über den M_IK_TE_X-Paketmanager hinzugefügt werden.

Das Installationsskript scheitert außerdem bei einigen Anwendern – aufgrund eingeschränkter Nutzerzugriffsrechte – beim Eintragen der Schriften in die Map-Datei. Dies muss gegebenenfalls durch den Anwender über die Kommandozeile `initexmf --edit-config-file updmap` erfolgen. In der sich öffnenden Datei sollte sich der Eintrag `Map tudscr.map` befinden. Ist dies nicht der Fall, muss diese Zeile manuell eingetragen und die Datei anschließend gespeichert werden. Danach muss der Nutzer in der Kommandozeile noch `initexmf --mkmaps` ausführen.

ANMERKUNG ZU T_EX LIVE UND M_AC_TE_X

Für den Schriftinstallationsprozess werden die Pakete **fontinst**, **fontware** sowie **cmbright**, **hfbright**, **cm-super**, **lmodern** und **iwona** benötigt. Sollte keine Vollinstallation von T_EX Live durchgeführt worden sein, müssen diese Pakete sehr aller Voraussicht nach manuell durch den Benutzer über den T_EX Live-Paketmanager hinzugefügt werden.

Sind nach einem fehlerfreien Durchlauf des Installationsskript die Schriften dennoch nicht verfügbar, so lässt sich mit `updmap-sys --syncwithtrees` die Synchronisierung aller Schriftdateien anstoßen. Daran anschließend muss mit `updmap-sys --enable Map=tudscr.map` die nun – hoffentlich – bekannte Map-Datei und mit `updmap-sys --force` die dazugehörigen Schriftdateien registriert werden.

Sind die Schriften danach immer noch nicht verfügbar, so wurden bestimmt schon weitere Schriften auf dem System *lokal* installiert. In diesem Fall sollte der Vorgang nochmals für eine lokale Schriftinstallation mit `updmap --syncwithtrees`, `updmap --enable Map=tudscr.map` und `updmap --force` ausgeführt werden. Dieses Vorgehen macht allerdings den Befehl `updmap-sys` von nun an wirkungslos. Nach einer systemweiten Installation neuer Schriften – beispielsweise bei der Aktualisierung der Distribution – müssen diese über den manuellen Aufruf von `updmap` zukünftig durch den Anwender lokal bei T_EX Live respektive M_AC_TE_X registriert werden.

Achtung!

Für die Schriftinstallation werden die Skripte *tftopl*, *pltotf* und *vptovf* benötigt, welche bei T_EX Live beziehungsweise M_AC_TE_X über das Paket **fontware** aus **collection-fontutils** bereitgestellt werden und zwingend installiert sein müssen.

WEITERFÜHRENDE INSTALLATIONSHINWEISE

In [Anhang A](#) sind zusätzliche Varianten der Installation von TUD-KOMA-Script zu finden. Soll eine ältere lokale Nutzerinstallation entfernt werden, um zukünftig alle Aktualisierungen direkt über die jeweils verwendete Distribution durchzuführen, werden entsprechende Skripte zu Deinstallation in [Unterkapitel A.1](#) zur Verfügung gestellt.

Will der Anwender *bewusst* eine lokale Nutzerinstallation anlegen, werden hierfür in [Unterkapitel A.2](#) entsprechende Skripte bereitgestellt. Soll eine bereits vorhandene, lokale Nutzerinstallation aktualisiert werden, finden Sie die entsprechenden Skripte in [Unterkapitel A.3](#).

Für die portablen Varianten von **T_EX Live Portable** respektive **MiK_TE_X Portable** sind zusätzliche Installationshinweise in [Unterkapitel A.4](#) zu finden.

1.2.1 INSTALLATION DER POSTSCRIPT-SCHRIFTEN UNTER WINDOWS

Zur Installation der Schriften des Corporate Designs für das TUD-KOMA-Script-Bundle ist das Archiv **TUD-KOMA-Script_fonts_Windows.zip**¹⁷ vorgesehen. Dieses ist sowohl für **T_EX Live** als auch **MiK_TE_X** nutzbar und enthält – bis auf die Schriftarchive selbst – alle benötigten Dateien. Diese sollten nach dem Entpacken des Archivs in das gleiche Verzeichnis kopiert werden. Vor der Verwendung des Skripts **tudscrfonts_install.bat** sollte sichergestellt werden, dass sich *alle* der folgenden Dateien im selben Verzeichnis befinden:

tudscrfonts_install.bat	Installationsskript
Univers_PS.zip	Archiv mit Schriftdateien für Univers
DIN_Bd_PS.zip	Archiv mit Schriftdateien für DIN BOLD
tudscrfonts.zip	Archiv mit Metriken für die Schriftinstallation via fontinst

Beim Ausführen des Installationsskripts werden alle Schriften standardmäßig in ein lokales Nutzerverzeichnis installiert. Wird das Skript über das Kontextmenü mit Administratorrechten ausgeführt, erfolgt die Installation in einem Pfad, der *für alle Nutzer* gültig und lesbar ist.

1.2.2 INSTALLATION DER POSTSCRIPT-SCHRIFTEN UNTER LINUX UND OS X

Für die Erstellung des Installationsskripts für Linux und OS X geht mein Dank an Jons-Tobias Wamhoff, der sich für die erstmalige Portierung des Skripts von Windows zu unixartigen Systemen freiwillig zur Verfügung stellte. Zur Installation der Schriften des Corporate Designs für das TUD-KOMA-Script-Bundle ist das Archiv **TUD-KOMA-Script_fonts_Unix.zip**¹⁸ vorgesehen. Dieses ist sowohl für **T_EX Live** als auch **Mac_TE_X** nutzbar und enthält – bis auf die Schriftdateien selbst – alle benötigten Dateien. Diese sollten nach dem Entpacken des Archivs in das gleiche Verzeichnis kopiert werden. Vor der Verwendung des Skripts **tudscrfonts_install.sh** sollte sichergestellt werden, dass sich *alle* der folgenden Dateien im selben Verzeichnis befinden:

tudscrfonts_install.sh	Installationsskript (Terminal: <code>bash tudscrfonts_install.sh</code>)
Univers_PS.zip	Archiv mit Schriftdateien für Univers
DIN_Bd_PS.zip	Archiv mit Schriftdateien für DIN BOLD
tudscrfonts.zip	Archiv mit Metriken für die Schriftinstallation via fontinst

Achtung!

Das Installationsskript ***muss*** mit `bash <Skript>.sh` im Terminal aufgerufen werden. Dabei werden alle Schriften standardmäßig in das lokale Nutzerverzeichnis (`$TEXMFHOME`) installiert. Wird das Skript mit `sudo` verwendet, erfolgt die Installation *für alle Nutzer* in den lokalen Systempfad (`$TEXMFLOCAL`).

1.2.3 PROBLEME BEI DER INSTALLATION DER POSTSCRIPT-SCHRIFTEN

Treten bei der Installation wider Erwarten Probleme auf, so sollte eine Logdatei erstellt werden. Unter Windows muss das Skript, welches Probleme verursacht, aus der Kommandozeile mit `<Skript>.bat > <Skript>.log` aufgerufen werden, wobei das doppelte Betätigen der Eingabetaste nötig ist. Für unixartige Systeme ist der Aufruf `bash <Skript>.sh > <Skript>.log`

¹⁷https://github.com/tud-cd/tudscr/releases/download/fonts/TUD-KOMA-Script_fonts_Windows.zip

¹⁸https://github.com/tud-cd/tudscr/releases/download/fonts/TUD-KOMA-Script_fonts_Unix.zip

aus dem Terminal heraus zu verwenden. Die erstellte Logdatei kann mit einer kurzen Fehlerbeschreibung direkt an tudscr@gmail.com gesendet werden.

1.3 SCHNELLEINSTIEG

Das Handbuch gliedert sich in drei Teile. In [Teil I](#) ist die Dokumentation von TUD-KOMA-Script zu finden. Hier werden alle neuen Optionen, Umgebungen und Befehle, die über die Funktionalität von KOMA-Script hinausgehen, erläutert. [Teil II](#) enthält zum einen einfache Minimalbeispiele, um den prinzipiellen Umgang und die Funktionalitäten von TUD-KOMA-Script zu demonstrieren. Zum anderen werden hier auch ausführliche und dokumentierte Tutorials vor allem für \LaTeX -Neulinge angeboten. Insbesondere das Tutorial [treatise.pdf](#) ist mehr als einen Blick wert, wenn eine wissenschaftliche Arbeit mit $\text{\LaTeX}2_{\epsilon}$ verfasst werden soll. Abschließend werden verschiedenste Pakete vorgestellt, die nicht speziell für das TUD-KOMA-Script-Bundle selber sondern auch für andere \LaTeX -Klassen verwendet werden können und demzufolge für alle \LaTeX -Anwender interessant sein könnten. Außerdem werden hier einige Tipps & Tricks beim Umgang mit \LaTeX beschrieben, um kleinere oder größere Probleme zu lösen.

Die Klassen [tudscrbook](#), [tudscrreprt](#) und [tudscrartcl](#) sind Wrapper-Klassen der bekannten KOMA-Script-Klassen [scrbook](#), [scrreprt](#) sowie [scrartcl](#) und können einfach anstelle deren verwendet werden. Auf diesen basierende Dokumente können durch das Umstellen der Dokumentklasse einfach in das Corporate Design der Technischen Universität Dresden überführt werden. Bei Fragestellungen bezüglich Layout, Schriften oder ähnlichem ist in jedem Fall ein weiterer Blick in das hier vorliegende Handbuch empfehlenswert.



TEIL I

DAS TUD-KOMA-SCRIPT-BUNDLE

Dies ist der Hauptteil des TUD-KOMA-Script-Bundles. Hier findet der Anwender alle verfügbaren Optionen, Umgebungen und Befehle, die über die Funktionalität von KOMA-Script hinausgehen.

KAPITEL 2

DIE HAUPTKLASSEN

Es werden die drei neuen Hauptklassen

tudscrbook
tudscrrept
tudscrartcl

eingeführt, welche auf den KOMA-Script-Klassen basieren und grundsätzlich alle deren Optionen, Umgebungen und Befehle – beispielsweise `parskip` für die Absatzeinstellungen oder `BCOR` zur Festlegung der Bindekorrektur – unterstützen. Zusätzlich zu den KOMA-Script-Klassen werden weitere Pakete zwingend benötigt, welche unter [Unterkapitel 6.1](#) aufgeführt sind und auf jeden Fall durch TUD-KOMA-Script geladen werden.

Es sei hier abermals auf die [Anwenderdokumentation von KOMA-Script](#)¹⁹ hingewiesen, viele der folgend beschriebenen Befehle und Optionen beziehen sich auf die darin vorgestellten Einstellungsmöglichkeiten. Die Anpassungen und Erweiterungen der KOMA-Script-Klassen an das Corporate Design und die neu definierten beziehungsweise geänderten Befehle und Optionen werden im Folgenden erläutert.

```
\TUDOptions{<Optionenliste>}  
\TUDOption{<Option>}{<Werteliste>}
```

Mit diesen Befehlen hat man bei den meisten der neuen Klassenoptionen die Möglichkeit, den Wert der Optionen noch nach dem Laden der Klasse zu ändern. Man kann wahlweise mit der Anweisung `\TUDOptions` die Werte einer Reihe von Optionen ändern. Jede Option der Optionenliste hat dabei die Form `<Option>=<Wert>`. Die meisten Optionen besitzen auch einen Säumniswert²⁰. Versäumt man die Angabe eines Wertes – verwendet demzufolge einfach die Form `<Option>` – so wird automatisch dieser Säumniswert angenommen.

Manche Optionen können gleichzeitig mehrere Werte besitzen. Für diese besteht die Möglichkeit, mit `\TUDOption` der einen Option nacheinander eine Reihe von Werten zuzuweisen. Die einzelnen Werte sind dabei in der Werteliste durch Komma voneinander getrennt.

Mit diesen beiden Befehlen kann im Bedarfsfall das Verhalten von einer Option oder mehreren Optionen im Dokument geändert werden. Werden diese Befehle in einer Umgebung oder einer Gruppe verwendet, bleiben die gemachten Einstellungen innerhalb dieser lokal begrenzt.

2.1 DIE SCHRIFTEN DES CORPORATE DESIGNS

Das Corporate Design der Technischen Universität Dresden gibt die Verwendung der Schriften **Univers** für den Fließtext sowie **DIN BOLD** für das Setzen von Überschriften vor, was durch TUD-KOMA-Script in der Standardkonfiguration auch so umgesetzt wird. Da jedoch in längeren Texten die Verwendung von Serifenschriften zu empfehlen ist, gibt es die Möglichkeit, die ursprünglich vorgesehenen Schriften des Corporate Designs nicht zu laden und stattdessen die \LaTeX -Standardschriften beziehungsweise ein anderes Schriftpaket zu verwenden. Die Erläuterungen dazu sind in [Abschnitt 2.1.1](#) zu finden.

Durch das Corporate Design werden keine Schriften für den Mathematiksatz festgelegt. Das ist insbesondere für sowohl mathematische Abhandlungen als auch ingenieur- und naturwissenschaftliche Dokumente nicht tragbar. Dieser Mangel wird behoben, indem im Mathematikmodus die lateinischen Buchstaben der Hausschriften mit griechischen Lettern und mathematischen

¹⁹<http://mirrors.ctan.org/macros/latex/contrib/koma-script/doc/scrguide.pdf>

²⁰engl.: default value

Symbolen aus anderen Paketen ergänzt werden.²¹ Diese Grundeinstellung lässt sich ebenfalls deaktivieren, wodurch die Standardschriften oder gegebenenfalls die eines zusätzlich geladenen Paketes für den mathematischen Satz genutzt werden. Die dafür relevanten Einstellungen werden in [Abschnitt 2.1.2](#) erläutert. In [Unterkapitel 5.8](#) sowie [Unterkapitel 5.9](#) sind zusätzliche Hinweise zum typografisch guten Mathematiksatz zu finden.

2.1.1 SCHRIFTEN FÜR DEN TEXTSATZ

v2.02

`cdfont=<Einstellung>` `preset: true`

Mit der Option `cdfont` können durch den Anwender alle zentralen Schrifteinstellungen für die TUD-KOMA-Script-Klassen vorgenommen werden. Dies betrifft sowohl die Schriften für Überschriften als auch den Fließtext sowie die Mathematikschriften. Die verwendete Schriftstärke im charakteristischen Querbalken der Kopfzeile lässt sich hiermit ebenfalls einstellen.

`false/off/no:`

Es werden die L^AT_EX-Standardschriften und nicht die Hausschriften des Corporate Designs verwendet. Der Anwender kann beliebige Schriftpakete nutzen.²² Sollte das Layout des Corporate Designs aktiviert sein (*siehe* `cd`), werden die Überschriften in Großbuchstaben und **DIN BOLD** gesetzt und nur die Schriftart des Fließtextes kann angepasst werden.

`true/on/yes/light/lightfont/noheavyfont:` (Säumniswert)

Es werden die Hausschriften im Stil des Corporate Designs der Technischen Universität Dresden genutzt. Überschriften der obersten Gliederungsebenen bis einschließlich `\subsubsection` verwenden **DIN BOLD**, darunter liegende²³ **Univers 65 Bold**. Für den Fließtext im Dokument kommt **Univers 45 Light** zum Einsatz. Aus `lmodern` wird die Schreibmaschinenschrift verwendet.

`heavy/heavyfont:`

Die Schriftstärke der Hausschriften wird erhöht. Die beiden untersten Gliederungsebenen werden in **Univers 75 Black** gesetzt, der Fließtext in **Univers 55 Regular**. Ansonsten entspricht alles der Option `cdfont=true`. Die Mathematikschriften werden nicht beeinflusst, diese lassen sich gegebenenfalls mit `\boldmath` auf den fetten Schnitt umschalten.

`nodin:`

Für die Überschriften wird **DIN BOLD** nicht verwendet. Für `cdfont=true` wird **Univers** genutzt. Die Schriftstärke ist dabei abhängig von der Einstellung `cdfont=light/heavy`. Ist die Verwendung der Schriften des Corporate Designs deaktiviert (`cdfont=false`), kommt die fette Schriftstärke der eingestellten serifenlosen Schriftfamilie zum Einsatz.

`din:`

Mit dieser Einstellung wird die Schrift **DIN BOLD** in den Überschriften verwendet. Diese ist standardmäßig aktiviert.

v2.02

Für den Text im Querbalken gibt es folgende Einstellmöglichkeiten:

`head/lighthead/lightfonthead/noheavyfonthead:`

Für den Querbalken der Kopfzeile wird unabhängig von der Verwendung der Hausschriften die Schrift **Univers** in normaler Schriftstärke verwendet, *siehe* `cdhead=true`.

`heavyhead/heavyfonthead:`

Die im Querbalken verwendete Schrift ist **Univers** in erhöhter Stärke, *siehe* `cdhead=heavy`.

²¹`jwona` für die Schrift **DIN BOLD** und zusätzlich `cmbright` für die **Univers**-Schriftfamilie

²²Für die Verwendung der klassischen L^AT_EX-Schriften, ist das Paket `lmodern` sehr empfehlenswert.

²³`\paragraph` und `\subparagraph`

Die verwendeten Mathematikschriften lassen sich mit folgenden Werte beeinflussen:

`nomath/nocdmath:`

Diese Einstellung deaktiviert die Verwendung von serifenlosen Schriften für den mathematischen Satz. Es werden die \LaTeX -Standardschriften verwendet und der Benutzer kann beliebige Schriftpakete für den Mathematikmodus nutzen, *siehe* `cdmath=false`.

`math/cdmath:`

Es werden serifenlose Mathematikschriften für lateinische und griechische Lettern genutzt, *siehe* `cdmath=true`.

`upgreek/uprightgreek:`

Die großen griechischen Buchstaben werden im Mathematikmodus aufrecht gesetzt, *siehe* `slantedgreek=false`.

`slgreek/slantedgreek:`

In mathematischen Umgebungen erfolgt die Ausgabe der griechischen Majuskeln kursiv, *siehe* `slantedgreek=true`.

v2.02
v2.04

Normalerweise kommen die Schriften des Corporate Designs im PostScript-Format zum Einsatz, wenn diese wie unter [Unterkapitel 1.2](#) beschrieben installiert wurden. Wird entweder \LuaTeX oder \XeLaTeX als Dokumentprozessor verwendet und das Paket `fontspec` geladen, so werden die Schriften des Corporate Designs im OpenType-Format verwendet. Hierfür sind die Hinweise in [Abschnitt 2.1.3 auf Seite 21](#) unbedingt zu beachten.

AUSZEICHNUNGEN IN ÜBERSCHRIFTEN

Für die Schriftauswahl der Überschriften aller Gliederungsebenen sind die durch KOMA-Script bereitgestellten Schriftelemente verantwortlich. Mehr dazu ist in [Abschnitt 2.2.8](#) zu finden. Da die Überschriften der obersten Gliederungsebenen bis einschließlich `\subsubsection` normalerweise in Majuskeln gesetzt werden, bestehen für den Anwender mit den folgenden Befehlen gewisse Einflussmöglichkeiten, deren Ausprägung anzupassen.

```
\ifdin{<Dann-Teil>}{<Sonst-Teil>}
```

Der Befehl `\ifdin` prüft, ob die Schriftfamilie **DIN BOLD** aktiv ist und führt in diesem Fall `{<Dann-Teil>}` aus, andernfalls `{<Sonst-Teil>}`. Dies ist beispielsweise bei Überschriften sinnvoll, wenn innerhalb des obligatorischen Argumentes zwischen der Ausgabe im Dokument selber und dem Eintrag für das Inhaltsverzeichnis sowie der Ausprägung der automatischen Kolumnentitel unterschieden werden soll.

```
\MakeTextUppercase{<Text>}  
\NoCaseChange{<Text>}
```

Der Befehl `\MakeTextUppercase` stammt aus dem Paket `textcase` und setzt den Text seines Argumentes in Majuskeln. Die Überschriften der Gliederungsebenen bis einschließlich `\subsubsection` werden damit in Großbuchstaben der Schrift **DIN BOLD** gesetzt, wenn das Layout des Corporate Designs nicht deaktiviert wurde (`cd=false`). Sollen in einer Überschrift bestimmte Kleinbuchstaben erhalten bleiben, ist der Befehl `\NoCaseChange` zu nutzen, welcher ebenfalls von besagtem Paket bereitgestellt wird.

Beispiel: In einer Kapitelüberschrift wird ein einzelnes Wort in Kleinbuchstaben geschrieben:

```
\chapter{Überschrift mit \NoCaseChange{kleinem} Wort}
```

AUSZEICHNUNGEN IM TEXT

Unabhängig davon, welche Schriftfamilie verwendet wird, können die Schriften des Corporate Designs jederzeit entweder mit einem der hier aufgeführten Textschalter oder Textkommandos innerhalb des Dokumentes genutzt werden. Ein Textschalter wirkt sich – wenn er nicht in einer Gruppe oder einer Umgebung verwendet und damit lokal begrenzt wird – global auf das Dokument aus, wie etwa beispielsweise `\bfseries`. Bei einem Textkommando hingegen erfolgt die Änderung der Schriftart nur für das nachfolgend angegebene Argument, wie zum Beispiel bei `\textbf{<Text>}`. Darauf ist bei der Nutzung zu achten.

```
\univln
\textuln{<Text>}
\univrn
\texturn{<Text>}
\univbn
\textubn{<Text>}
\univxn
\textuxn{<Text>}
\univls
\textuls{<Text>}
\univrs
\texturs{<Text>}
\univbs
\textubs{<Text>}
\univxs
\textuxs{<Text>}
\dinbn
\textdbn{<Text>}
```

Schriftart	Schalter	Textkommando
Univers 45 Light	<code>\univln</code>	<code>\textuln{<Text>}</code>
Univers 55 Regular	<code>\univrn</code>	<code>\texturn{<Text>}</code>
Univers 65 Bold	<code>\univbn</code>	<code>\textubn{<Text>}</code>
Univers 75 Black	<code>\univxn</code>	<code>\textuxn{<Text>}</code>
<i>Univers 45 Light Oblique</i>	<code>\univls</code>	<code>\textuls{<Text>}</code>
<i>Univers 55 Regular Oblique</i>	<code>\univrs</code>	<code>\texturs{<Text>}</code>
<i>Univers 65 Bold Oblique</i>	<code>\univbs</code>	<code>\textubs{<Text>}</code>
<i>Univers 75 Black Oblique</i>	<code>\univxs</code>	<code>\textuxs{<Text>}</code>
DIN BOLD	<code>\dinbn</code>	<code>\textdbn{<Text>}</code>

Die Schrift **DIN BOLD** darf laut Corporate Design nur mit Majuskeln (Großbuchstaben) verwendet werden. Wird diese Schrift manuell verwendet, sollte dies mit `\MakeTextUppercase{\textdbn{<Text>}}` geschehen. Sollen dabei im Argument einzelne Teile zwingend klein geschrieben werden, wird der Befehl `\NoCaseChange` benötigt.

Alternativ zu den beschriebenen Textschaltern und -kommandos können seit der Version v2.04 auch die beiden Befehle `\cdfont` und `\textcdfont` verwendet werden, welche die gleiche Funktionalität wesentlich komfortabler bereitstellen.

v2.04

```
\cdfont{<Schriftart>}
\textcdfont{<Schriftart>}{<Text>}
```

Diese beiden Befehle dienen ebenfalls zu gezielter Aktivierung einer Schriftart des Corporate Designs in Stärke und Schnitt. Hierbei entspricht `\cdfont` einem Textschalter und ändert die aktuell verwendete Schriftart im aktuellen Geltungsbereich auf `<Schriftart>`, wohingegen `\textcdfont` als Textkommando fungiert und den im zweiten Argument gegebenen `<Text>` in `<Schriftart>` setzt ohne dabei die Dokumentschriftart selbst zu ändern.

Für die Schriftauswahl muss im ersten Argument der Name der zu verwendenden Schriftart angegeben werden. Dieser ist der obigen Tabelle zu entnehmen. Für die Auswahl der Schriftfamilie **Univers** kann der Vorsatz `Univers` im Argument `<Schriftart>` entfallen. Ebenso sind weder Leerzeichen noch die passende Groß- und Kleinschreibung notwendig. Für die Wahl der Schriftstärke ist die entsprechende Zahl *oder* die Bezeichnung allein ausreichend.²⁴ Anstelle des Suffix' *Oblique* ist auch die Nutzung von *Italic* oder *Slanted* als Alias für die geeigneten Schriftschnitte möglich. Zur Auswahl von **DIN BOLD** ist `din` als Argument hinreichend.

²⁴45/55/65/75 oder Light/Regular/Bold/Black

2.1.2 SCHRIFTEN FÜR DEN MATHEMATIKSATZ

v2.03

`cdmath=<Ein-Aus-Wert>`

`preset: true | cdfont=false: false`

Diese Option dient zur Anpassung der Mathematikschriften. Wird diese aktiviert, so werden zu den Hausschriften passende im Mathematikmodus genutzt, mit `cdmath=false` wird auf die Standardschriften zurückgeschaltet. Ein Umschalten innerhalb des Dokumentes ist – beispielsweise für Abbildungen oder Tabellen – durch `\TUDoptions{cdmath=true/false}` möglich. Mit `\boldmath` kann auf fette Mathematikschriften umgeschaltet werden. Gültige Werte für die Option `cdmath` sind:

`false/off/no:`

Es werden die normalen \LaTeX -Serifenschriften beziehungsweise die Schriften beliebig nutzbarer Pakete für den Mathematiksatz verwendet.

`true/on/yes:` (Säumniswert)

Im Mathematikmodus wird **Univers** genutzt. Außerdem kommen die griechischen Buchstaben aus dem Paket `cmbright` sowie Symbole aus dem Paket `iwona` zum Einsatz.

`upgreek/uprightgreek:`

Die griechischen Majuskeln werden aufrecht gesetzt, *siehe* `slantedgreek=false`.

`slgreek/slantedgreek:`

Die Ausgabe der griechischen Großbuchstaben erfolgt kursiv, *siehe* `slantedgreek=true`.

GRIECHISCHEN BUCHSTABEN

`\varDelta`
`\varTheta`
`\varLambda`
`\varXi`
`\varPi`
`\varSigma`
`\varUpsilon`
`\varPhi`
`\varPsi`
`\varOmega`
`\upDelta`
`\upTheta`
`\upLambda`
`\upXi`
`\upPi`
`\upSigma`
`\upUpsilon`
`\upPhi`
`\upPsi`
`\upOmega`

Befehl (kursiv)	Symbol	Befehl (aufrecht)	Symbol
<code>\varDelta</code>	Δ	<code>\upDelta</code>	Δ
<code>\varTheta</code>	Θ	<code>\upTheta</code>	Θ
<code>\varLambda</code>	Λ	<code>\upLambda</code>	Λ
<code>\varXi</code>	Ξ	<code>\upXi</code>	Ξ
<code>\varPi</code>	Π	<code>\upPi</code>	Π
<code>\varSigma</code>	Σ	<code>\upSigma</code>	Σ
<code>\varUpsilon</code>	Υ	<code>\upUpsilon</code>	Υ
<code>\varPhi</code>	Φ	<code>\upPhi</code>	Φ
<code>\varPsi</code>	Ψ	<code>\upPsi</code>	Ψ
<code>\varOmega</code>	Ω	<code>\upOmega</code>	Ω

Die Befehle `\up<Name>` und `\var<Name>` werden normalerweise durch einige Pakete, unter anderem auch von `cmbright` oder `amsmath`, bereitgestellt.

Unabhängig von den beiden Optionen `cdmath` und `slantedgreek` können sowohl kursive als auch aufrechte griechischen Großbuchstaben im Mathematikmodus mit diesen Befehlen direkt verwendet werden. Dies ist nützlich, um zwischen kursiven Variablen und aufrechten Konstanten zu unterscheiden. Die griechischen Minuskeln sind leider nur in der kursiven Variante verfügbar.

```
slantedgreek=<Ein-Aus-Wert>
```

```
preset: false
```

Die Option ändert die standardmäßige Neigung der griechischen Großbuchstaben im Mathematikmodus bei der Verwendung der Befehle `\Delta`, `\Theta`, `\Lambda`, `\Xi`, `\Pi`, `\Sigma`, `\Upsilon`, `\Phi`, `\Psi` und `\Omega`. Wie unabhängig von der Option `slantedgreek` gezielt kursive und aufrechte Buchstaben gesetzt werden können, wird auf der vorherigen Seite beschrieben.

`false/off/no`:

Die griechischen Majuskeln werden wie bei den Standardklassen aufrecht gesetzt.

`true/on/yes`: (Säumniswert)

Die Ausgabe der griechischen Großbuchstaben erfolgt kursiv.

ZUSÄTZLICHE HINWEISE ZUM MATHEMATIKSATZ

Weitere Hinweise zum typografisch guten Mathematiksatz sind außerdem in [Unterkapitel 5.9](#) sowie [Unterkapitel 5.8](#) zu finden.

2.1.3 DIE SCHRIFTEN DES CORPORATE DESIGNS IM OPENTYPE-FORMAT

v2.02

Das TUD-KOMA-Script-Bundle unterstützt die Nutzung der Schriften des Corporate Designs sowohl im PostScript- als auch im OpenType-Format. Letztere müssen ebenfalls über das [Universitätsmarketing auf Anfrage](#)²⁵ bestellt werden. Die in den beiden Archiven `Univers_8_TTF.zip` und `DIN_TTF.zip` enthaltenen Schriften müssen für das Betriebssystem installiert werden und lassen sich anschließend mit dem Paket `fontspec` verwenden.

Auf die Installation der PostScript-Schriften kann dennoch nicht ohne Weiteres verzichtet werden. Denn einerseits sind diese für die Kompilierung eines \LaTeX -Dokumentes über den klassischen Prozesspfad via `latex` \rightarrow `dvips` \rightarrow `ps2pdf` – wie es beispielsweise für die Erstellung von Grafiken mit `pstricks` notwendig ist – nötig. Andererseits liefern die Schriftfamilien des Corporate Designs keinerlei mathematische Glyphen, sodass diese bei der PostScript-Schriftinstallation aus den Schriftpaketen `cmbright` und `iwona` entnommen werden. Bei der Nutzung der Schriften im OpenType-Format ist dies leider nicht so einfach möglich, da es schlichtweg an passenden Schriftpaketen für den Mathematiksatz im OpenType-Format mangelt. Weiterhin kommt es auch beim Kerning der Schriften zu Problemen.

Die Verwendung der Schriften des Corporate Designs im OpenType-Format sollte folglich nur erfolgen, wenn eine Installation der PostScript-Schriften *absolut* nicht möglich ist beziehungsweise \LuaTeX oder \XeTeX zwingend genutzt werden müssen. Hierfür ist es – nach der systemweiten Installation der OpenType-Schriften – ausreichend, das Paket `fontspec` zu laden.

2.2 DAS LAYOUT DES CORPORATE DESIGNS

Das Hauptaugenmerk der neuen Klassen liegt auf der Umsetzung des Corporate Designs der Technischen Universität Dresden für \LaTeX . Ein großer Teil der definierten Optionen und Befehle dient genau dazu und wird nachfolgend beschrieben.

Einige spezielle Seiten werden im prägnanten Stil mit dem Logo der Technischen Universität Dresden und der dazugehörigen Kopfzeile mit Querbalken gesetzt. Dies betrifft insbesondere [die Umschlagseite und den Titel aus Abschnitt 2.2.3](#), die [Teileseiten in Abschnitt 2.2.4](#) sowie die [Kapitelseiten in Abschnitt 2.2.5](#). Mit den `tudheadings`-Seitenstilen oder der `tudpage`-Umgebung

²⁵http://tu-dresden.de/service/publizieren/cd/1_basiselemente/03_hausschrift/schriftbestellung.html

können weitere Seiten in diesem Stil erzeugt werden. Wird das Paket **tudscrsvsupervisor** verwendet und mit den bereitgestellten Befehlen oder Umgebungen eine Aufgabenstellung, ein Gutachten oder ein Aushang erstellt, so erscheinen auch diese in besagtem Seitenstil.

2.2.1 DIE GESTALT VON TITEL, UMSCHLAGESEITE, TEILEN UND KAPITELN

v2.03
v2.04

`cd=<Einstellung>` `preset: true`

Mit dieser Option wird festgelegt, ob und wie das Corporate Design der Technischen Universität Dresden verwendet wird. Sie hat Einfluss auf die Ausprägung von Titel-, Teil-, und Kapitelseiten sowie die Überschriften der weiteren Gliederungsebenen.

`false/off/no:`

Diese Einstellung erzeugt das Standard-Verhalten der KOMA-Script-Klassen, es wird kein Corporate Design genutzt.

`true/on/yes/nocolor/monochrome:` (Säumniswert)

Das Layout für Titel-, Teil- und Kapitelseiten ist im Corporate Design, es wird schwarze Schrift für Titel, Teil- und Kapitelüberschriften verwendet. Die Ausprägung des Seitenkopfes ist abhängig von der Option `cdhead`.

`lightcolor/pale:`

Die Einstellung entspricht weitestgehend der Option `cd=true`, allerdings wird die primäre Hausfarbe **HKS41** für den Kopf des `tudheadings`-Seitenstils und die Überschriften genutzt.

v2.04

`barcolor:`

Zusätzlich zur vorherigen Einstellung wird außerdem der Querbalken farblich abgesetzt.

v2.03

`bicolor/bichrome:`

Der Kopf wird mit einem farbigen Hintergrund in der Hausfarbe gesetzt, auch der Querbalken wird farbig hinterlegt. Für die Überschriften wird die primären Hausfarbe verwendet.

`color:`

Der Titel sowie Teil- und Kapitelseiten werden allesamt farbig und im Corporate Design gestaltet, der Seitenkopf wird in der primären Hausfarbe **HKS41** gesetzt, der Querbalken erhält Linien als Begrenzung.

v2.03

`full/fullcolor:`

Entspricht der vorherigen Einstellung, allerdings wird der Querbalken nicht durch Linien begrenzt sondern farbig hinterlegt.

EINSTELLUNGEN FÜR TITEL, UMSCHLAGESEITE, TEILE UND KAPITEL

Das Verhalten aller für das Layout relevanten Elemente wird von der eben zuvor erläuterten Option `cd=<Einstellung>` bestimmt. Dies betrifft zum einen sowohl den Titel (`\maketitle`) als auch die Umschlagseite (`\makecover`) und zum anderen alle Teileseiten (`\part`, `\addpart`) und Kapitelseiten (`\chapter`, `\addchap`).

Soll ein bestimmtes Element des Layouts abweichend erscheinen, so kann eine der folgenden Optionen genutzt werden, um dieses individuell anzupassen. Die gültigen Wertzuweisungen für die einzelnen Elemente entsprechend dabei den möglichen Werten für die Option `cd`.

v2.03
v2.04

`cdtitle=<Einstellung>`

Mit der Option `cdtitle` kann die allgemeine Einstellung für den Titel überschrieben werden. Es kann zwischen dem normalen (`cdtitle=false`) und dem im Corporate Design umgeschaltet

werden. Die neue Titelseite unterstützt alle durch KOMA-Script definierten Befehle für den Titel.²⁶ Zusätzlich werden viele neue Felder definiert, welche vor allem für eine wissenschaftliche Arbeit von Relevanz sind. Genauer dazu ist in [Abschnitt 2.2.3](#) nachzulesen. Unabhängig von der gewählten Variante der Titelseite wird diese immer mit `\maketitle` erzeugt.

v2.02
v2.03
v2.04

`cdcover=<Einstellung>`

Die TUD-KOMA-Script-Klassen führen zusätzlich den Befehl `\makecover` ein, mit dem sich neben dem Titel eine separate Umschlagseite erzeugen lässt. Diese ist in ihrer Gestalt der Titelseite sehr ähnlich, wird normalerweise jedoch in einem anderen Satzspiegel als dem des Buchblocks gesetzt. Mit der Option `cdcover` kann – unabhängig von `cd` – das Erscheinungsbild der Umschlagseite geändert werden. Wird `cdcover=false` gewählt, entspricht die Umschlagseite dem originalen KOMA-Script-Titel. Die Verwendung des Befehls `\makecover` sowie die dazugehörigen Parameter werden detailliert in [Abschnitt 2.2.3](#) erläutert.

v2.03
v2.04

`cdpart=<Einstellung>`

Für die Teileseiten kann der Wert des Schlüssels `cd` separat überschrieben und somit deren Layout respektive Erscheinungsbild beeinflusst werden, welches bei der Benutzung der Befehle `\part` beziehungsweise `\addpart` und deren Sternversionen genutzt wird.

v2.03
v2.04

`cdchapter=<Einstellung>`

Für Kapitelseiten kann der Schlüsselwert `cd` ebenfalls angepasst und damit das Layout respektive Erscheinungsbild geändert werden, das bei der Verwendung von `\chapter` beziehungsweise `\addchap` und den dazugehörigen Sternversionen genutzt wird.

v2.05

`cdsection=<Einstellung>`

Für Überschriften der Gliederungsebenen `\section`, `\subsection`, `\subsubsection` sowie `\paragraph` und `\subparagraph` werden in der primären Hausfarbe [HKS41](#) ausgegeben, falls über die Option `cd` eine farbige Ausprägung des Layouts eingestellt wurde. Mit der Angabe von `cdsection=true` erschienen die Überschriften der genannten Gliederungsebenen ohne Farbeinsatz.

Beispiel: Soll die Titelseite in Farbe, der Rest des Dokumentes allerdings in schwarzer Schrift gesetzt werden, so kann dies folgendermaßen erreicht werden:

```
\documentclass[cd=true,cdtitle=color]{<Dokumentklasse>}
```

POSITION VON ÜBERSCHRIFTEN

v2.02

`\pageheadingsvskip` (Länge)
`\headingsvskip` (Länge)

Mit einer Änderung dieser beiden Längen kann die vertikale Position bestimmter Überschriften verändert werden. Sowohl der Titel auf einer Titelseite (KOMA-Script-Option `titlepage=true`) als auch die Überschriften von Teilen und separaten Kapitelseiten (`chapterpage=true`) lassen sich mit `\pageheadingsvskip` vertikal verschieben. Demgegenüber erlaubt es die Länge

²⁶`\extratitle{<Schmutztitel>}`, `\titlehead{<Kopf>}`, `\subject{<Typisierung>}`, `\title{<Titel>}`, `\subtitle{<Untertitel>}`, `\author{<Autor>}`, `\date{<Datum>}`, `\publishers{<Verlag>}`, `\and` und `\thanks{<Fußnote>}` sowie `\uppertitleback{<Titelrückseitenkopf>}`, `\lowertitleback{<Titelrückseitenfuß>}` und `\dedication{<Widmung>}`

`\headingsvskip`, sowohl den Titel eines Titelpfotes (KOMA-Script-Option `titlepage=false`) als auch die Kapitelüberschrift bei deaktivierter Kapitelseite (`chapterpage=false`) in ihrer vertikalen Position anzupassen.

Die zuvor genannten Überschriften werden normalerweise im Layout relativ tief im Textbereich gesetzt. Mit negativen Werten werden die Überschriften nach oben verschoben, wobei darauf geachtet werden sollte, dass diese sich danach noch innerhalb des Satzspiegels befinden. Positive Werte setzen die Überschriften dementsprechend tiefer.

2.2.2 SEITEN IM STIL DES CORPORATE DESIGNS

v2.02

`tudheadings` (Seitenstil)
`plain.tudheadings` (Seitenstil)
`empty.tudheadings` (Seitenstil)

Ein zentrales Element des Corporate Designs der Technischen Universität Dresden ist der prägnante Seitenkopf mit der Angabe von Fakultät (`\faculty`), Einrichtung (`\department`), Institut (`\institute`) und Lehrstuhl (`\chair`) im dazugehörigen Querbalken. Durch die Nutzung des Paketes `scrlayer-scrpage` lassen sich entweder einzelne Seiten oder auch ganze Dokumente sehr einfach in diesem Stil setzen. Hierzu muss lediglich mit `\pagestyle{<Seitenstil>}` einer der Seitenstile geladen werden.

Achtung!

Allen Seitenstilen gemein ist der typische Kopf mit dem charakteristischen Querbalken, dessen Gestalt für *alle* Seitenstile gleichermaßen über die Option `cdhead` angepasst werden kann. Mit dem Befehl `\headlogo` lässt sich ein zusätzliches Zweitlogo im Kopfbereich ausgegeben. *Für die speziellen Layout-Elemente Titel und Umschlagseite sowie Teile- und Kapitelseiten wird die Einstellung von `cdhead` durch die Nutzung der Option `cd` überschrieben.*

Die Ausprägung des Fußes unterscheidet sich bei den einzelnen Seitenstilen. Dieser ist beim Seitenstil `empty.tudheadings` immer leer. Die beiden Stile – oder vielmehr das Seitenstil-Paar – `tudheadings` und `plain.tudheadings` übernehmen die Einstellungen für die Fußzeile aus der Anwenderschnittstelle von `scrlayer-scrpage`.²⁷ Wie die einzelnen Befehle zur Individualisierung der Fußzeile zu verwenden sind, kann der **KOMA-Script**=**Anleitung**²⁸ entnommen werden. Alternativ zu einer eigenen Definition der Fußzeile lässt sich außerdem die Option `cdfoot` verwenden. Zusätzlich kann über `\footcontent` ein freier Inhalt für den Fußbereich definiert werden, mit `\footlogo` ist die Ausgabe von einem oder mehreren Logos in diesem möglich. Die verwendete Schrift im Fußbereich wird durch das Schriftelement `tudheadings` festgelegt.

Sobald einer der definierten Stile mit `\pagestyle{<Seitenstil>}` aktiviert wurde, sind die beiden Seitenstile `tudheadings` sowie `plain.tudheadings` zusätzlich unter den Namen `headings` respektive `plain` verwendbar. Dies hat den Vorteil, dass bei Optionen oder Befehlen, welche automatisch zwischen den beiden Seitenstilen `headings` und `plain` umschalten, durch die einmalige Auswahl von einem der `tudheadings`-Stilen nun zwischen diesen umgeschaltet wird.

Der Seitenstil `empty` erzeugt allerdings weiterhin eine komplett leere Seite. Soll eine Seite mit der prägnanten Kopfzeile der Technischen Universität Dresden und leerem Seitenfuß erscheinen, so muss `\pagestyle{empty.tudheadings}` manuell aufgerufen werden. Um auf das normale Verhalten von KOMA-Script zurückzuschalten, muss einer der beiden Stile `scrheadings` beziehungsweise `plain.scrheadings` aktiviert werden.

Achtung!

²⁷Es können die Befehle `\lefoot`, `\cefoot` und `\refoot` sowie `\lofoot`, `\cofoot` und `\rofoot` respektive `\ofoot`, `\cfoot` und `\ifoot` genutzt werden.

²⁸<http://mirrors.ctan.org/macros/latex/contrib/koma-script/doc/scrguide.pdf>

Die beschriebenen Seitenstile werden erst nach dem Laden des Paketes `scrlayer-scrpage` definiert. Wird dieses nicht durch den Anwender geladen, sollte `\pagestyle{<Seitenstil>}` erst nach `\begin{document}` verwendet werden.

v2.04

tudheadings (Schriftelement)

Im Fußbereich der Seiten im `tudheadings`-Seitenstil wird das Schriftelement `tudheadings` verwendet. Dieses wirkt sich auf die Seitenzahlen, den Kolumnentitel und die mit `\footcontent` angegebenen Inhalte aus. Hierüber wird die Wahl der richtigen Schriftfarbe in Abhängigkeit vom Seitenhintergrund und den Einstellungen für die Optionen `cdhead` sowie `cdfoot` realisiert. Wie `tudheadings` angepasst werden kann, ist in [Abschnitt 2.2.8](#) zu finden.

```
\faculty{<Fakultät>}
\department{<Einrichtung>}
\institute{<Institut>}
\chair{<Lehrstuhl>}
\extraheadline{<Textzeile>}
```

Für den Seitenstil des Corporate Designs der Technischen Universität Dresden typisch ist die Kopfzeile mit dem charakteristischen Querbalken. In dieser wird – falls angegeben – in fetter Schrift die Fakultät ausgegeben, danach folgen durch Kommas getrennt die Einrichtung, das Institut und der Lehrstuhl beziehungsweise die Professur. Sollte der Platz in der ersten Zeile nicht ausreichen, erfolgt ein automatischer Zeilenumbruch.

In besonderen Ausnahmefällen erlaubt das Corporate Design die Angabe einer zusätzlichen zweiten beziehungsweise dritten Zeile unterhalb der Angaben des Bereichs an der Technischen Universität Dresden, welche weitere, frei wählbare Angaben enthält. Diese kann mit dem Befehl `\extraheadline{<Textzeile>}` definiert werden.

v2.03

v2.04

`cdhead=<Einstellung>` `preset: true`

Mit dieser Option lassen sich für die `tudheadings`-Seitenstile sowohl die Gestalt des Logos sowie des Querbalkens als auch die darin verwendete Schrift beeinflussen. Die folgenden Werte können für eine Anpassung der Schriftart im Balken verwendet werden:

`false/off/no`:

Sollte mit `cdfont=false` die Verwendung der Hausschrift im Stil des Corporate Designs der Technischen Universität Dresden deaktiviert worden sein, wird die Kopfzeile im Querbalken in den Serifenlosen der genutzten Schrift gesetzt. Sind die Hausschriften aktiviert, hat diese Einstellung keinen Einfluss.

`true/on/yes/light/lightfont/noheavyfont`: (Säumniswert)

Im Querbalken wird für `\faculty` **Univers 65 Bold** verwendet, die Felder `\department`, `\institute`, `\chair` und `\extraheadline` kommt **Univers 45 Light** zum Einsatz.

`heavy/heavyfont`:

Der Inhalt von `\faculty` wird weiterhin in **Univers 65 Bold** gesetzt, für die restlichen Felder wird **Univers 55 Regular** genutzt.

Bei der Ausprägung des Kopfes und des Querbalkens gibt es mehrere Varianten. Einerseits kann der Querbalken mit zwei Außenlinien dargestellt werden:

`nocolor/monochrome`:

Der Kopf und die Linien des Querbalkens erscheinen in schwarzer Farbe.

`lightcolor/pale`:

Sowohl Kopf als auch Querbalken werden in der primären Hausfarbe gesetzt.

Andererseits ist auch eine Darstellung mit mehr Farbeinsatz möglich, bei welcher der Querbalken und gegebenenfalls der ganze Seitenkopf farblich abgesetzt wird.

v2.04

`barcolor:`

Im Gegensatz zur vorherigen Einstellung wird der Querbalken mit farbigem Hintergrund verwendet.

v2.03

`bicolor/bichrome:`

Die Kopfzeile wird farblich abgesetzt, wobei der Hintergrund des Logos und der Querbalken unterschiedlich ausfallen. Die Außenlinien der Querbalkens entfallen.

Für den Fall, dass der Querbalken lediglich mit zwei Außenlinien dargestellt wird, kann zusätzlich dessen Laufweite festgelegt werden:

`textwidth/slim:`

Der Querbalken im Kopf erstreckt sich nur über den Textbereich. Diese Einstellung ist insbesondere sinnvoll, wenn ein randloser Ausdruck technisch nicht möglich ist.

`paperwidth/wide:`

Die horizontale Ausdehnung des Querbalkens erstreckt sich über die komplette Seitenbreite bis an den Blattrand. Dieses Verhalten ist standardmäßig im farbigen Layout aktiviert.

v2.03

`cdfoot=<Einstellung>`

`preset: false`

Die TUD-KOMA-Script-Klassen sind – insbesondere aufgrund der Möglichkeit zur Verwendung des Paketes `scrlayer-scrpage` – bei der Gestaltung der Kopf- und Fußzeilen sehr flexibel und individuell anpassbar. Die Ausprägung und der Inhalt dieser ist nicht explizit durch das Corporate Design vorgegeben und können durch den Anwender beliebig gewählt und geändert werden.

Eine Möglichkeit für deren Gestaltung zeigt das Handbuch für das Corporate Design der Technischen Universität Dresden. Dieses wird ohne Kopf- und mit einer einfachen Fußzeile gesetzt, welche den aktuellen Kolumnentitel sowie die Paginierung enthält. Mit `cdfoot` kann diese Ausprägung aktiviert werden, was auch für dieses Anwenderhandbuch geschehen ist.

`false/off/no:`

Die Kopf- und Fußzeilen zeigen Standardverhalten, zur manuellen Änderung sollte unbedingt das KOMA-Script-Paket `scrlayer-scrpage` verwendet werden.

`true/on/yes:` (Säumniswert)

Die Fußzeilen des Dokumentes werden wie im Handbuch des Corporate Designs der Technischen Universität Dresden mit Kolumnentitel und Seitenzahl gesetzt. Im doppelseitigen Satz (Klassenoption `twoside=true`) wird die Paginierung außen platziert.

Wird beim Laden der Klasse respektive des Paketes `scrlayer-scrpage` die KOMA-Script-Option `manualmark` nicht explizit angegeben, so werden für `cdfoot=true` über die KOMA-Script-Option `automark` auch gleichzeitig die automatischen Kolumnentitel aktiviert, welche als Marken die Titel der Gliederungsebenen verwendet. Genauereres hierzu sowie der Möglichkeit, die Kolumnentitel manuell festzulegen, ist dem [KOMA-Script-Handbuch](#)²⁹ zu entnehmen.

v2.03

Sollte einer der `tudheadings`-Seitenstil aktiviert sein und es wird auf der erzeugten Seite ein farbiges Layout – beispielsweise der zweifarbige Kopf (`cdhead=bicolor`) oder ein farbiger Seitenhintergrund – genutzt, so kann auch die Fußzeile einen farbigen Hintergrund erhalten.

`nocolor/monochrome:`

Der Fuß wird immer ohne farbigen Hintergrund gesetzt.

²⁹<http://mirrors.ctan.org/macros/latex/contrib/koma-script/doc/scrguide.pdf>

color/bicolor/bichrome:

Die Fußzeile wird farblich abgesetzt, falls entweder der zweifarbige Kopf (`cdhead=bicolor`) oder zumindest eine Seite mit einem farbigen Hintergrund in der Hausfarbe (Titel oder Kapitelseite) verwendet wurde.

Wird der Option `cdfoot` eine Längenmaß übergeben, entspricht dies der Verwendung von `extrabottommargin`.

`\headlogo[<Parameterliste>]{<Dateiname>}`

Neben dem Logo der Technischen Universität Dresden darf zusätzlich ein Zweitlogo im Kopf verwendet werden. Dieses lässt sich mit diesem Befehl einbinden. Normalerweise wird es auf die Höhe der Erstlogos skaliert. Über das optionale Argument können weitere Formatierungsbeefehle an den verwendeten Befehl `\includegraphics` durchgereicht werden, um beispielsweise die Größe des Zweitlogos anzupassen. Welche Parameter angepasst werden können, ist der Dokumentation des `graphicx`-Paketes zu entnehmen.

Sollte die Option `ddc` aktiviert sein, wird das DRESDEN-concept-Logo nicht im Kopf sondern automatisch im Fuß gesetzt. Die Option `ddchead` setzt dieses auf jeden Fall im Kopf und überschreibt damit das mit `\headlogo` angegebene Zweitlogo.

v2.03

`\footlogo[<Parameterliste>]{<Dateinamenliste>}`
`\footlogosep`
`\footlogoheight` (Länge)

Laut den Richtlinien des Corporate Designs dürfen im Fußbereich weitere Logos erscheinen, beispielsweise von kooperierenden Unternehmen oder Sponsoren. Die Dateinamen der gewünschten Logos können als kommagetrennte Liste im obligatorischen Argument des Befehls `\footlogo` angegeben werden. Sollte tatsächlich nicht nur ein Dateiname sondern eine Liste übergeben worden sein, so wird bei der Ausgabe der Logos zwischen diesen jeweils der in `\footlogosep` gespeicherte Separator – standardmäßig `\hfill` – gesetzt. Dieser kann mit `\renewcommand*\footlogosep{...}` beliebig durch den Anwender angepasst werden. Der Separator wird auch gesetzt, wenn in der `{<Dateinamenliste>}` lediglich ein Komma verwendet wurde. So kann man beispielsweise ein Logo mit `\footlogo{,<Dateiname>,}` zentriert im Fuß setzen.

Das optionale Argument von `\footlogo` wird an `\includegraphics` weitergereicht. Dies geschieht für alle angegebenen Dateien aus der Liste gleichermaßen. Sollen für einzelne Logos individuelle Einstellungen vorgenommen werden, so sind die entsprechenden Parameter im obligatorischen Argument nach dem jeweiligen Dateinamen mit einem Doppelpunkt „:“ als Separator (`\footlogo{<Dateiname>:<Parameter>}`) zu übergeben, wobei diese *nach* den allgemeinen Einstellungen für alle Logos angewendet werden. Die möglichen Parameter und Werte für die optionalen Argumente sind der Dokumentation des `graphicx`-Paketes zu entnehmen.

Ohne die Angabe eines optionalen Argumentes für die Größe werden alle Logos im Fuß auf die auf die Höhe des Logos der Technischen Universität Dresden skaliert. Der Anwender kann dies ändern, indem der Wert der Länge `\footlogoheight` mit `\setlength` auf einen von 0pt verschiedenen gesetzt wird. Sollte die Höhe des Fußbereiches nicht ausreichen, um alle Logos in der gewünschten Größe darstellen zu können, kann diese über die Option `extrabottommargin` beziehungsweise `cdfoot` angepasst werden.

```
\footcontent [<Anweisungen>]{<Inhalt>}[<Inhalt>]
\footcontent* [<Anweisungen>]{<Inhalt>}[<Inhalt>]
```

Mit diesem Befehl kann beliebiger Inhalt entweder einspaltig oder zweispaltig im Fußbereich der `tudheadings`-Seitenstile gesetzt werden. In der Form `\footcontent{<Inhalt>}` wird der Inhalt über die komplette Textbreite im Fuß ausgegeben. Wird der Befehl jedoch in der zweiten Variante `\footcontent{<linker Inhalt>}[<rechter Inhalt>]` mit einem optionalen *nach* dem obligatorischen Argument verwendet, so erscheint der Fußbereich zweispaltig, wobei der Inhalt aus dem ersten, obligatorischen Argument in der linken und der Inhalt aus dem zweiten, optionalen Argument entsprechend in der rechten Fußspalte gesetzt wird.

Im Normalfall wird das Schriftelement `tudheadings` für die Schrift im Fußbereich verwendet. Mit dem ersten optionalen Argument können weitere Schrifteinstellungen respektive Anweisungen vor der eigentlichen Ausgabe des Inhaltes erfolgen. Soll die Definition des Inhalts für den Fußbereich gänzlich ohne die automatische Schriftformatierungen erfolgen, so kann die Sternversion `\footcontent*` genutzt und gegebenenfalls die Schriftformatierung über das optionale Argument vorgenommen werden.

<code>ddc=<Einstellung></code>	<code>preset: false</code>
<code>ddchead=<Einstellung></code>	<code>preset: false</code>
<code>ddcfoot=<Einstellung></code>	<code>preset: false</code>

Diese Optionen fügen das Logo von DRESDEN-concept entweder im Kopf oder im Fuß der Seiten mit dem Stil `tudheadings` ein. Mit `ddc` wird dieses automatisch entweder im Kopf oder – falls ein Zweitlogo mit `\headlogo` angegeben wurde – im Fuß gesetzt. Die anderen beiden Optionen setzen das Logo zwingend entweder im Kopf (`ddchead`) oder im Fuß (`ddcfoot`), wobei erstgenannte ein optionales Zweitlogo dabei unterdrückt. Die Verwendung einer der drei Optionen führt zur Deaktivierung der anderen beiden, sie schließen sich folglich gegenseitig aus. Die möglichen Werte für diese Optionen sind:

`false/off/no`:

Bei den `tudheadings`-Seitenstile erscheint kein Logo von DRESDEN-concept.

`true/on/yes`: (Säumniswert)

Das Logo von DRESDEN-concept wird im Kopf beziehungsweise im Fuß verwendet. Die Wahl der Farbe des Logos geschieht passend zur farblichen Ausprägung der Seite selbst.

Soll die Farbe des DRESDEN-concept-Logos manuell erfolgen, können folgende Werte verwendet werden:

`color`:

Im Kopf oder Fuß wird die achtfarbige 4C-Variante des DRESDEN-concept-Logos genutzt.

`colorblack`:

Es wird das achtfarbige Logo mit schwarzem DRESDEN-concept-Schriftzug anstelle des grauen verwendet. Für den Fuß wird der grüne Claim ebenfalls durch einen schwarzen ersetzt. Dies ist insbesondere für kleine Darstellungen des Logos im Fuß sinnvoll.

`gray/grey`:

Dies Ausgabe des DRESDEN-concept-Logos erfolgt in Graustufen.

`black`:

Verwendung des Logos in Graustufen mit schwarzem Schriftzug.

`blue`:

Der Schriftzug und das Logo werden in der primären Hausfarbe `HKS41` und den entsprechenden Abstufungen gesetzt

white:

Das DRESDEN-concept-Logo sowie der dazugehörige Schriftzug sind vollständig weiß. Die Größe des DRESDEN-concept-Logos ist vorgegeben. Es wird sowohl im Kopf als auch im Fuß in der gleichen Höhe gesetzt, wie das Logo der Technischen Universität Dresden. Wird es im Fuß gesetzt, lässt sich die Größe allerdings über die Länge `\footlogoheight` ändern. Sollte nach einer Vergrößerung der Darstellung die Höhe des Fußbereiches nicht ausreichen, so kann diese über die Option `cdfoot` beziehungsweise `extrabottommargin` angepasst werden.

v2.02
v2.03

```
\begin{tudpage}[<Sprache> || <Parameterliste>]
...
\end{tudpage}
language=<Sprache> (Parameter)
columns=<Anzahl> (Parameter)
pagestyle=<Einstellung> (Parameter)
cdfont=<Einstellung> (Parameter)
cdhead=<Einstellung> (Parameter)
cdfoot=<Einstellung> (Parameter)
headlogo=<Dateiname> (Parameter)
footlogo=<Dateinamenliste> (Parameter)
ddc=<Einstellung> (Parameter)
ddchead=<Einstellung> (Parameter)
ddcfoot=<Einstellung> (Parameter)
```

(siehe `cdfont` auf Seite 17)

(siehe `cdhead` auf Seite 25)

(siehe `cdfoot` auf Seite 26)

(siehe `\headlogo` auf Seite 27)

(siehe `\footlogo` auf Seite 27)

(siehe `ddc` auf Seite 28)

(siehe `ddchead` auf Seite 28)

(siehe `ddcfoot` auf Seite 28)

Die `tudpage`-Umgebung hat ihren Ursprung in einer früheren Version, als die `tudheadings`-Seitenstile noch nicht verfügbar waren, welche mittlerweile anstelle dieser Umgebung verwendet werden können. Für die `tudpage`-Umgebung lassen sich verschiedene Parameter als optionales Argument angegeben. Wird das Paket `babel` genutzt, kann die verwendete Sprache mit `language=<Sprache>` geändert werden, was zur Anpassung der sprachspezifischen Trennungsmuster und Bezeichner führt. Wurde das Paket `multicol` geladen, wird mit dem Parameter `columns=<Anzahl>` der Inhalt der Umgebung mehrspaltig gesetzt. Mit `pagestyle` kann der Seitenstil angepasst werden, wobei `headings`, `plain` und `empty` gültige Werte sind.

Die weiteren Parameter entsprechen in ihrem Verhalten prinzipiell den gleichnamigen Klassenoptionen respektive Befehlen, wirken sich jedoch nur lokal innerhalb der `tudpage`-Umgebung aus. Das Verhalten sowie die jeweils gültigen Wertzuweisungen können in den entsprechenden Abschnitten des Handbuchs nachgelesen werden.

2.2.3 DER TITEL UND DIE UMSCHLAGESEITE

Für den Titel werden alle Felder unterstützt, die bereits durch KOMA-Script bereitgestellt werden. Darüber hinaus werden für die TUD-KOMA-Script-Klassen weitere Felder definiert, die Auswirkungen auf die Gestalt des Titels haben. Diese werden nachfolgend in diesem Abschnitt erläutert. Der Titel – bestehend aus möglichem Schmutztitel, der eigentlichen Titelseite und der nachgelagerten Elementen – kann mit dem Befehl `\maketitle` ausgegeben werden. Außerdem kann im zweispaltigen Satz der `\maketitleonecolumn` verwendet werden, welcher einen einspaltigen Einfügung nach dem Titel selbst ermöglicht.

Zusätzlich zum Titel lässt sich mit `\makecover` eine Umschlagseite erzeugen. Diese kann insbesondere für gebundene Arbeiten verwendet werden. Es wird – im Vergleich zum Titel – lediglich einer reduzierte Anzahl an Feldern auf dieser ausgegeben.

v2.02

Für alle Felder des Titels und der Umschlagseite lässt sich die verwendete Schrift anpassen. Dabei werden sowohl die bereits durch KOMA-Script bereitgestellten Schriftelemente `titlehead`, `subject`, `title`, `subtitle`, `author`, `date`, `publishers` und `dedication` als auch die neuen `titlepage` und `thesis` unterstützt.

Beispiel: In diesem Dokument wurde der Untertitel derart geändert, dass dieser nicht standardmäßig in **DIN BOLD** sondern in **Univers 65 Bold** ausgegeben wird.

```
\addtokomafont{subtitle}{\univbn}
```

v2.01

v2.02

v2.03

```
\maketitle[<Seitenzahl> || <Parameterliste>]
```

```
pagenumber=<Seitenzahl> (Parameter)
```

```
cdfont=<Einstellung> (Parameter)
```

(siehe `cdfont` auf Seite 17)

```
cdhead=<Einstellung> (Parameter)
```

(siehe `cdhead` auf Seite 25)

```
cdfoot=<Einstellung> (Parameter)
```

(siehe `cdfoot` auf Seite 26)

```
headlogo=<Dateiname> (Parameter)
```

(siehe `\headlogo` auf Seite 27)

```
footlogo=<Dateinamenliste> (Parameter)
```

(siehe `\footlogo` auf Seite 27)

```
ddc=<Einstellung> (Parameter)
```

(siehe `ddc` auf Seite 28)

```
ddchead=<Einstellung> (Parameter)
```

(siehe `ddchead` auf Seite 28)

```
ddcfoot=<Einstellung> (Parameter)
```

(siehe `ddcfoot` auf Seite 28)

Der Befehl `\maketitle` setzt für `cdtitle=false` den normalen KOMA-Script-Titel, ansonsten wird die Titelseite im Corporate Design der Technischen Universität Dresden erzeugt. Die letztere Variante ist im Vergleich zum Standardtitel um eine Vielzahl von Feldern erweitert worden und erlaubt insbesondere die Angabe von Daten für das Deckblatt einer akademischen Abschlussarbeit. Die einzelnen Felder werden später in diesem Abschnitt erläutert. Wird das Dokument doppelseitig und mit rechts öffnenden Kapiteln gesetzt,³⁰ so wird zusätzlich die Option `clearcolor` beachtet. Dies gilt es insbesondere bei der Verwendung der Befehle `\uppertitleback` respektive `\lowertitleback` für die Titelseite zu beachten.

Das optionale Argument erlaubt – ebenso wie bei den KOMA-Script-Klassen – die Änderung der Seitenzahl der Titelseite. Diese wird jedoch nicht ausgegeben, sondern beeinflusst lediglich die Zählung. Sie sollten hier unbedingt eine ungerade Zahl wählen, da sonst die gesamte Zählung durcheinander gerät. Wird eine Titelseite (KOMA-Script-Option `titlepage=true`) im Corporate Design der Technischen Universität Dresden gesetzt (`cdtitle=true`), können auch die weiterhin aufgeführten Parameter im optionalen Argument verwendet werden. Diese entsprechen in ihrem Verhalten den gleichnamigen Optionen respektive Befehlen, wirken sich jedoch nur lokal und einzig auf die Titelseite aus. So kann beispielsweise die Nutzung eines DRESDEN-concept-Logos auf den Titel beschränkt bleiben.

```
\maketitleonecolumn[<Seitenzahl> || <Parameterliste>]{<Einspaltentext>}
```

Im zweispaltigen Satz (Klassenoption `twocolumn=true`) wird mit `\maketitle` die Titelseite selbst immer einspaltig gesetzt. Direkt nach dem Titel folgt normalerweise der zweispaltige Fließtext. Mit dem Befehl `\maketitleonecolumn` kann nach dem Titel zusätzlich weiterer Inhalt – zum Beispiel eine Zusammenfassung respektive eine Kurzfassung – einspaltig gesetzt werden.

Bei einer aktivierten Titelseite (KOMA-Script-Option `titlepage=true`) erfolgt die Ausgabe des Argumentes (`<Einspaltentext>`) direkt nach dieser auf einer oder gegebenenfalls mehreren neuen Seiten ebenfalls einspaltig. Wird der Befehl `\maketitleonecolumn` statt mit einer

³⁰KOMA-Script-Optionen `twoside` und `open=right`, Standard für `tudscrbook`

Titelseite jedoch mit einem Titelpopf (KOMA-Script-Option `titlepage=false`) zum Einsatz, so folgt diesem die einspaltige Textpassage aus dem obligatorischen Argument direkt. Dabei erfolgt gegebenenfalls ein automatischer Seitenumbruch, falls der Inhalt nicht auf eine einzelne Seite passt. Nach dem obligatorischen Argument `<Einspaltentext>` wird direkt und ohne zusätzlichen Seitenumbruch auf das zweispaltige Layout umgeschaltet.

Der optionale Parameter von `\maketitleonecolumn` kann äquivalent zu `\maketitle` für die Änderung der Seitenzahl, der verwendeten Schrift sowie zur Anpassung von Kopf und Fuß verwendet werden. Dabei ist zu beachten, dass ein Großteil der Parameter nur Auswirkungen haben, falls eine Titelseite (KOMA-Script-Option `titlepage=true`) verwendet wird.

v2.02
v2.03

```
\makecover[<Seitenzahl> || <Parameterliste>]
pagenumber=<Seitenzahl> (Parameter)
cdgeometry=<Ein-Aus-Wert> (Parameter)
cdfont=<Einstellung> (Parameter)
cdhead=<Einstellung> (Parameter)
cdfoot=<Einstellung> (Parameter)
headlogo=<Dateiname> (Parameter)
footlogo=<Dateinamenliste> (Parameter)
ddc=<Einstellung> (Parameter)
ddchead=<Einstellung> (Parameter)
ddcfoot=<Einstellung> (Parameter)
```

(siehe `cdfont` auf Seite 17)

(siehe `cdhead` auf Seite 25)

(siehe `cdfoot` auf Seite 26)

(siehe `\headlogo` auf Seite 27)

(siehe `\footlogo` auf Seite 27)

(siehe `ddc` auf Seite 28)

(siehe `ddchead` auf Seite 28)

(siehe `ddcfoot` auf Seite 28)

Eine Umschlagseite wird zumeist für gebundene Abschlussarbeiten verlangt, um diese beispielsweise für einen Prägedruck auf dem Buchdeckel zu verwenden. Deshalb ist die farbige Ausprägung der Umschlagseite auch deaktiviert, wenn diese für das restliche Dokument aktiv ist (`cd=color`). Dies kann jedoch jederzeit mit `cdcover=<Einstellung>` überschrieben werden.

Wird `cdcover=true` gewählt, so wird die Umschlagseite im Corporate Design der Technischen Universität Dresden gesetzt. Auf dieser werden der Titel des Dokumentes, die Typisierung durch `\thesis` und/oder `\subject` sowie der Autor oder respektive die Autoren und gegebenenfalls der mit `\publishers` angegebene Verlag ausgegeben. Für die Einstellung `cdcover=false` wird lediglich der normale KOMA-Script-Titel als separate Umschlagseite ausgegeben.

v2.02

Die Titelseite selbst gehört immer zum Buchblock und wird daher im gleichen Satzspiegel gesetzt. Dem entgegen steht die Umschlagseite, welche zumeist in einem anderen Layout erscheint. Normalerweise wird das Cover – unabhängig von der Option `cdgeometry` – im asymmetrischen Satzspiegel des Corporate Designs gesetzt. Mit `cdgeometry=false` im optionalen Argument kann das Verhalten geändert werden. In diesem Fall erscheint auch die Umschlagseite im Buchblock des restlichen Dokumentes. Allerdings können für diese Einstellung die Seitenränder mit den Befehlen `\coverpagetopmargin`, `\coverpageleftmargin`, `\coverpagerightmargin` sowie `\coverpagebottommargin` durch den Nutzer frei angepasst werden. Mehr dazu ist im **KOMA-Script-Handbuch**³¹ zu finden.

Außerdem kann mit dem optionalen Argument die Seitenzahl der Umschlagseite geändert werden. Diese wird jedoch nicht ausgegeben, sondern beeinflusst lediglich die Zählung. Sie sollten hier unbedingt eine ungerade Zahl wählen, da sonst die gesamte Zählung durcheinander gerät. Die weiterhin aufgeführten Parameter entsprechen in ihrem Verhalten beziehungsweise ihrer Funktion den gleichnamigen Optionen respektive Befehlen, wirken sich jedoch nur lokal und einzig auf die Umschlagseite aus.

³¹<http://mirrors.ctan.org/macros/latex/contrib/koma-script/doc/scrguide.pdf>

```
\titledelimiter{<Trennzeichen>}
```

Für den Titel und die Umschlagseite werden durch die TUD-KOMA-Script-Klassen eine Reihe von zusätzlichen Feldern bereitgestellt. Einigen dieser Felder wird eine Beschreibung (*siehe Unterkapitel 2.4*) vorangestellt. Dazwischen wird bei der Ausgabe ein Trennzeichen eingefügt. Ein Doppelpunkt gefolgt von einem Leerzeichen (`\nobreakspace`) ist hierfür die Voreinstellung. Mit dem Befehl `\titledelimiter` lässt sich dieses Trennzeichen beliebig an die individuellen Wünsche des Anwenders anpassen.

```
\author{<Autor(en)>}
\authormore{<Autorenzusatz>}
\dateofbirth{<Geburtsdatum>}
\placeofbirth{<Geburtsort>}
\matriculationnumber{<Matrikelnummer>}
\matriculationyear{<Immatrikulationsjahr>}
```

Mit dem Befehl `\author` wird der Autor angegeben. Innerhalb des Argumentes können auch mehrere Autoren aufgeführt werden, wobei diese in diesem Fall jeweils mit `\and` zu trennen sind. Alle weiteren hier vorgestellten Befehle können selbst im Argument von `\author` verwendet werden, wodurch für jeden Autor individuelle Angaben möglich sind.

Mit `\authormore` wird unter dem Autor eine Zeile ausgegeben, welche durch den Anwender frei belegt werden kann. Sollte das Paket `isodate` geladen sein, so wird die damit eingestellte Formatierung des Datums durch `\dateofbirth` – wie übrigens bei jedem anderem Datumsfeld der TUD-KOMA-Script-Klassen auch – verwendet. Dafür der Befehl `\printdate` aus diesem Paket verwendet. Die weiteren Befehle als zusätzliche Angabe erklären sich von selbst.

```
\and
```

Dieser Befehl wird sowohl bei den L^AT_EX-Standardklassen als auch bei den KOMA-Script-Klassen lediglich auf der Titelseite dazu verwendet, mehrere Autoren im Argument von `\author` voneinander zu trennen.

Bei den TUD-KOMA-Script-Klassen hingegen ist dieser Befehl derart in seiner Funktion erweitert worden, dass damit die Angabe einer kollaborativen Autorenschaft für Abschlussarbeiten innerhalb des Befehls `\author` möglich ist. Außerdem kann er noch im Argument von `\supervisor`, `\referee` sowie `\advisor` verwendet werden, um mehrere Betreuer beziehungsweise Gutachter und Fachreferenten anzugeben. Er ist dabei nicht auf die Verwendung für den Titel allein beschränkt sondern kann auch bei der Angabe von Personen in den entsprechenden Feldern der Umgebungen `task`, `evaluation` und `notice` aus dem Paket `tudscrsvsupervisor` eingesetzt werden.

Beispiel: Angenommen, es soll eine Abschlussarbeit von zwei unterschiedlichen Autoren in kollaborativer Gemeinschaft erstellt werden, so könnte man die Autorenangaben folgendermaßen gestalten:

```
\author{%
  Mickey Mouse
  \matriculationnumber{12345678}
  \dateofbirth{2.1.1990}
  \placeofbirth{Dresden}
\and%
  Donald Duck
```


Tabelle 2.1: Spezielle Werte zur Typisierung des Dokumentes für `\thesis` und `\subject`

Wert	Bezeichner	Deutsch	Englisch
diss	<code>\dissertationname</code>	Dissertation	Dissertation
doctoral	<code>\dissertationname</code>	Dissertation	Dissertation
phd	<code>\dissertationname</code>	Dissertation	Dissertation
diploma	<code>\diplomathesisname</code>	Diplomarbeit	Diploma Thesis
master	<code>\masterthesisname</code>	Master-Arbeit	Master Thesis
bachelor	<code>\bachelorthesisname</code>	Bachelor-Arbeit	Bachelor Thesis
student	<code>\studentresearchname</code>	Studienarbeit	Student Research Project
project	<code>\projectpapername</code>	Projektarbeit	Project Paper
seminar	<code>\seminarpapername</code>	Seminararbeit	Seminar Paper
research	<code>\researchname</code>	Forschungsbericht	Research Report
log	<code>\logname</code>	Protokoll	Log
report	<code>\reportname</code>	Bericht	Report
internship	<code>\internshipname</code>	Praktikumsbericht	Internship Report

```

\matriculationnumber{87654321}
\dateofbirth{1.2.1990}
\placeofbirth{Berlin}
}
\matriculationyear{2010}

```

Alle zusätzlichen Angaben außerhalb des Argumentes von `\author` werden für beide Autoren gleichermaßen übernommen. Angaben innerhalb des Argumentes von `\author` werden den jeweiligen, mit `\and` getrennten Autoren zugeordnet. Mehr dazu ist im Minimalbeispiel in [Unterkapitel 5.3](#).

```

\thesis{<Typisierung>}
\subject{<Typisierung>}

```

Mit diesen beiden Befehlen kann der Typ der Dokumentes beziehungsweise der Abschlussarbeit angegeben werden. Während der Befehl `\thesis` den Inhalt des Feldes unter dem Titel vertikal zentriert und in **DIN BOLD** auf der Titelseite ausgibt, erscheint der Inhalt des Befehls `\subject` in **Univers** oberhalb des Titels. Es können auch beide Befehle parallel mit unterschiedlichen Inhalten verwendet werden. Der Befehl `\thesis` dient den TUD-KOMA-Script-Dokumentklassen außerdem zur Erkennung von Abschlussarbeiten gedacht, da für diese spezielle Felder bereitgehalten werden und auch die Titelseite leicht geändert gesetzt wird.

Des Weiteren ist es bei beiden Befehlen möglich, spezielle Werte als Argument zur Typisierung des Dokumentes zu verwenden. Diese werden entsprechend der gewählten Dokumentensprache – entweder Deutsch oder Englisch – entschlüsselt und gesetzt. Die möglichen Werte sind [Tabelle 2.1](#) zu entnehmen. Dabei ist zu beachten, dass das Setzen eines speziellen Wertes für *entweder* `\thesis` oder `\subject` möglich ist. Die Verwendung eines der genannten Werte führt immer dazu, dass das Dokument als Abschlussarbeiten erkannt und die erweiterte Titelseite aktiviert wird. Gleichzeitig wird damit die Option `subjectthesis` beeinflusst. Sollte vom Anwender kein explizites Verhalten für `subjectthesis` definiert sein, so führt die Verwendung von `\thesis{<Wert>}` zu `subjectthesis=false` und `\subject{<Wert>}` zu `subjectthesis=true`.

```
subjectthesis=<Ein-Aus-Wert>
```

```
preset: false | \subject{<Tabelle 2.1>}: true
```

Der Befehl `\thesis` dient den TUD-KOMA-Script-Hauptklassen zur Unterscheidung zwei unterschiedlicher Ausprägungen der Titelseite und ist speziell für Abschlussarbeiten gedacht. Außerdem kann bei der Nutzung spezieller Werte aus [Tabelle 2.1](#) innerhalb des Argumentes von `\subject` ebenfalls das Verhalten für Abschlussarbeiten aktiviert werden, wobei hierdurch die Einstellung `subjectthesis=true` automatisch vorgenommen wird.

Für den Standardfall – bekanntlich `subjectthesis=false` – wird der durch `\thesis` gegebene Typ der Abschlussarbeit sowie der gegebenenfalls durch `\graduation` gesetzte angestrebte Abschluss in großen Lettern und sehr zentral auf der Titelseite gesetzt. Die Verwendung von `\subject` ist hierbei weiterhin möglich. Wird die Option mit `subjectthesis=true` aktiviert, so wird die mit `\thesis` gesetzte Bezeichnung nicht unterhalb sondern oberhalb des Titels an der Stelle von `\subject` ausgegeben. Der mit `\graduation` angegebene Abschluss wird weiterhin unter dem Titel, allerdings in schlankerer Schrift gesetzt. Eine etwaige Verwendung des Befehls `\subject` wird in diesem Fall ignoriert.

false/off/no:

Die Ausgabe des Typs der Abschlussarbeit (`\thesis`) selbst sowie des angestrebten Abschlusses (`\graduation`) erfolgt in großen Lettern in **DIN BOLD** zentral auf der Titelseite.

true/on/yes: (Säumniswert)

Der Typ der Abschlussarbeit (`\thesis`) wird oberhalb des Titels in der Betreffzeile gesetzt. Der angestrebte Abschluss (`\graduation`) wird zentral in der schlankeren **Univers** ausgegeben.

v2.02

```
\graduation[<Kurzform>]{<Grad>}
```

Mit diesem Befehl wird der angestrebte akademische Grad auf der Titelseite ausgegeben. Da dies nur mit einer Abschlussarbeit erreicht werden kann erfolgt die Ausgabe nur, wenn entweder `\thesis` oder `\subject` verwendet wurde, wobei bei letzterem Befehl im Argument zwingend ein Wert aus [Tabelle 2.1](#) verwendet werden muss.

Die Option `subjectthesis` hat Einfluss auf die Ausgabe auf der Titelseite. Für die Einstellung `subjectthesis=false` wird der Abschluss – ähnlich wie der Typ der Abschlussarbeit – zentral und in relativ großen Lettern gesetzt. Für `subjectthesis=true` erfolgt die Ausgabe kleiner und in weniger starken Buchstaben.

```
\supervisor{<Name(n)>}  
\referee{<Name(n)>}  
\advisor{<Name(n)>}  
\professor{<Name>}
```

Mit `\supervisor`, `\referee` und `\advisor` werden die Betreuer einer Abschlussarbeit beziehungsweise die Gutachter und Fachreferenten einer Dissertation angegeben. Zusätzlich kann mit `\professor` der betreuende Hochschullehrer beziehungsweise die betreuenden Professoren für studentische Arbeiten angegeben werden. Die Angabe mehrerer Person erfolgt wie beim Befehl `\author` durch die Trennung mittels `\and`.

```
\date[<Ergänzung>]{<Datum>}  
\defensedate{<Verteidigungsdatum>}
```

Mit `\date` kann das Datum angegeben werden. Das optionale Argument erlaubt eine zusätzliche Anmerkung, welche nach dem Datum ausgegeben wird. Das Datum wird bei normalen

Dokumenten direkt nach dem Autor beziehungsweise den Autoren ausgegeben. Bei Abschlussarbeiten – aktiviert durch die Verwendung von `\thesis` oder `subjectthesis` – erscheint dieses am Ende der Titelseite als Abgabedatum. Außerdem kann in diesem Fall mit dem Befehl `\defensedate` das Datum der Verteidigung angegeben werden, wie es beispielsweise bei dem Druck von Dissertationen üblich ist.

Sollte das Paket `isodate` geladen sein, so wird die damit eingestellte Formatierung des Datums durch den Befehl `\printdate` aus diesem Paket für alle Datumsfelder des Dokumentes und folglich auch für die beiden Felder `\date` und `\defensedate` verwendet.

v2.01

```
\extratitle{<Schmutztitel>}
\titlehead{<Kopf>}
\title{<Titel>}
\subtitle{<Untertitel>}
\publishers{<Verlag>}
\thanks{<Fußnote>}
\uppertitleback{<Titelrückseitenkopf>}
\lowertitleback{<Titelrückseitenfuß>}
\dedication{<Widmung>}
```

Diese Befehle entsprechen den in ihrem Verhalten den originalen Pendants der KOMA-Script-Klassen und sollen hier der Vollständigkeit halber erwähnt werden.

Die Ausgabe des mit `\extratitle` definierten Schmutztitels – welcher beliebig gestaltet und formatiert werden kann – erfolgt als Bestandteil der Titelei mit `\maketitle` vor der eigentlichen Titelseite. Mit dem Befehl `\titlehead` kann ein zusätzlicher, beliebig formatierbarer Text oberhalb der Typisierung und des Titels ausgegeben werden. Da die vertikale Position des Dokumenttitels durch das Corporate Design fest vorgegeben ist, kann es – im Gegensatz zu den KOMA-Script-Klassen – passieren, dass der Kopf des Haupttitels selbst in die Kopfzeile ragt. Dies wird durch die TUD-KOMA-Script-Klassen nicht geprüft und muss gegebenenfalls vom Anwender kontrolliert werden.

Die Befehle `\title` und `\subtitle` bedürfen keiner weiteren Erklärung. Anzumerken ist, dass sowohl Titel als auch Untertitel normalerweise in Majuskeln und **DIN BOLD** gesetzt werden. Der mit dem Befehl `\publishers` definierte Inhalt muss nicht zwingende einen Verlag bezeichnen sondern kann auch andere Informationen beinhalten, welche am Ende der Titelseite ausgegeben werden sollen.

Fußnoten werden auf dem Titel nicht mit `\footnote`, sondern mit der Anweisung `\thanks` erzeugt. Diese dienen in der Regel für Anmerkungen bei Titel oder den Autoren. Als Fußnotenzeichen werden dabei Symbole statt Zahlen verwendet. Der Befehl `\thanks` kann nur innerhalb des Arguments einer der Anweisungen für die Titelseite wie beispielsweise `\author` oder `\title` verwendet werden.

Im doppelseitigen Druck lässt sich die Rückseite der Haupttitelseite für weitere Angaben nutzen. Sowohl den Titelrückseitenkopf als auch den Titelrückseitenfuß kann der Anwender mit `\uppertitleback` und `\lowertitleback` frei gestalten.

Mit `\dedication` lässt sich eine separate Widmungsseite zentriert und in etwas größerer Schrift setzen. Die Rückseite ist – wie auch die des Schmutztitels – grundsätzlich leer. Die Widmung wird mit der restlichen Titelei ausgegeben und muss daher vor der Nutzung von `\maketitle` angegeben werden.

titlepage (Schriftelement)
thesis (Schriftelement)

Die TUD-KOMA-Script-Klassen definieren diese neuen Schriftelemente. Dabei wird `titlepage` auf der Titelseite für alle Felder verwendet, welche kein spezielles Schriftelement verwenden, welches ohnehin durch KOMA-Script bereitgestellt wird. Das mit `\thesis` angegebene Feld, in welchem der Typ einer Abschlussarbeit angegeben wird, nutzt das Schriftelement `thesis`. Wie diese Elemente angepasst werden können, ist in [Abschnitt 2.2.8](#) zu finden.

2.2.4 DIE TEILESEITE

Wird für die Teileseiten das Layout des Corporate Designs verwendet, so wird der Seitenstil dieser (`\partpagestyle`) auf `plain.tudheadings` gesetzt. Möchten Sie stattdessen einen anderen Seitenstil nutzen, so kann dieser mit `\renewcommand*\partpagestyle{<Seitenstil>}` angepasst werden.

`parttitle=<Ein-Aus-Wert>` `preset: false`

Diese Option ermöglicht es, den mit `\title` gegebenen Titel des Dokumentes selbst in großer Schrift auf einer Teileseite auszugeben, die Bezeichnung des mit `\part{<Bezeichnung>}` erzeugten Teils wird in diesem Fall in kleiner Schrift direkt darunter gesetzt. Diese Layout-Variante findet sich im Handbuch für das Corporate Design der Technischen Universität Dresden. Für die Klasse `tudscrartcl` ist diese Einstellung nicht verfügbar.

false/off/no:

Die Bezeichnung des Teils erscheint in großer Schrift auf der Seite, der Titel des Dokumentes gar nicht.

true/on/yes: (Säumniswert)

Der Titel wird in großer Auszeichnung auf der Teileseite gesetzt, die Bezeichnung des Teils selber in kleinerer.

parttitle (Schriftelement)

Mit dem Schriftelement `parttitle` lässt sich – bei aktivierter Option `parttitle` – die Schrift für die Bezeichnung des Teils beeinflussen. In [Abschnitt 2.2.8](#) ist zu finden, wie es angepasst werden kann.

2.2.5 DIE KAPITELSEITE

`chapterpage=<Ein-Aus-Wert>` `preset: false | cd=color: true`

Mit dieser Einstellung kann die Überschrift eines Kapitels separat auf einer Seite ausgegeben werden. Der nachfolgende Text wird auf der nächsten beziehungsweise bei doppelseitigem Satz und rechts öffnenden Kapiteln³² auf der übernächsten Seite ausgegeben. Die in diesem Fall erzeugte Rückseite wird in ihrer Ausprägung – wie auch Teileseiten – durch die Einstellung von `cleardoublespecialpage` bestimmt. Beim farbigen Layout ist diese Option standardmäßig aktiviert. Für die Klasse `tudscrartcl` ist diese Einstellung nicht verfügbar.

false/off/no:

Es gibt keine Sonderstellung von Kapiteln, der nachfolgende Text wird direkt unter der

³²KOMA-Script-Optionen `twoside` und `open=right`, Standard für `tudscrbook`

Überschrift respektive nach der mit `\setchapterpreamble` erzeugten Kapitelpräambel auf der gleichen Seite ausgegeben.

true/on/yes: (Säumniswert)

Die Kapitelüberschrift und gegebenenfalls die Kapitelpräambel werden auf einer separaten Seite gesetzt. Der folgende Text erscheint auf der nächsten respektive übernächsten Seite, *siehe auch* [cleardoublespecialpage](#).

v2.02

Mit `\renewcommand*{\chapterpagestyle}{<Seitenstil>}` lässt sich übrigens – unabhängig von der Option [chapterpage](#) – der Seitenstil von Kapiteln anpassen. Bei der Verwendung von separaten Kapitelseiten ist außerdem das Aktivieren der KOMA-Script-Option `chapterprefix` empfehlenswert. Damit werden die Kapitelüberschriften mit einer Vorsatzzeile gesetzt. Falls ein nummeriertes Kapitel erzeugt wird, so wird zunächst in einer Zeile „Kapitel“ gefolgt von der aktuellen Kapitelnummer ausgegeben, in der nächsten Zeile wird anschließend die eigentliche Überschrift in linksbündigem Flattersatz ausgegeben. Genauer hierzu ist in der [KOMA-Script=Dokumentation](#)³³ nachzulesen.

2.2.6 VAKATSEITEN

Automatisch erzeugte Vakatsseiten – auch absichtliche Leerseiten genannt – findet man am Beginn von Teilen und Kapiteln in Dokumenten mit den aktivierten KOMA-Script-Optionen `twoside` und `open=left` beziehungsweise `open=right`³⁴. Für diese kann der Seitenstil mit der KOMA-Script-Option `cleardoublepage` eingestellt werden.

<code>cleardoublespecialpage=<Einstellung></code>

`preset: true`

Diese Option wirkt sich lediglich bei aktiviertem doppelseitigem Satz und ausschließlich rechts eröffnenden Seiten für Teile beziehungsweise Kapitel aus.³⁵ In diesem Fall kann der Stil der darauffolgenden, linken Seite – sprich der Rückseite – beeinflusst werden. Das Normalverhalten sieht vor, dass nach einem Teil die Rückseite unabhängig von der Einstellung für `cleardoublepage` immer als vollständig leere Seite ohne Kopf- oder Fußzeilen gesetzt wird.

Diese Einstellung erlaubt es, dieses Normalverhalten zu deaktivieren und für die Seite nach der Teileseite – und abhängig von [chapterpage](#) auch nach einem Kapitelanfang auf einer separaten Seite – den Seitenstil der Option `cleardoublepage` zu übernehmen. Des Weiteren kann auch ein anderer, beliebiger, bereits definierter Seitenstil gewählt werden. Außerdem kann im farbigen Layout die Rückseite in der gleichen Farbe wie die Vorderseite von Teil oder Kapitel gesetzt werden. Für die Klasse [tudscrartcl](#) ist diese Einstellung nicht verfügbar.

false/off/no:

Die Rückseiten sind vollständig leere Seiten, unabhängig von Option `cleardoublepage`.

true/on/yes: (Säumniswert)

Der Seitenstil der Rückseite von Teilen und gegebenenfalls Kapiteln entspricht der Einstellung von `cleardoublepage` für Vakatsseiten.

current:

Es wird der aktuell definierte Seitenstil (`\pagestyle`) für die erzeugte Rückseite verwendet.

<Seitenstil>:

Mit der Angabe von `cleardoublespecialpage=<Seitenstil>` kann ein beliebiger, bereits definierter Seitenstil für die Rückseite nach Teilen und Kapiteln verwendet werden.

³³<http://mirrors.ctan.org/macros/latex/contrib/koma-script/doc/scrguide.pdf>

³⁴Standard bei [tudscrbook](#)

³⁵KOMA-Script-Option `twoside` und `open=right`

color:

Im farbigen Layout ist auch die Rückseite von Teilen und Kapiteln farbig, *siehe* [clearcolor](#).

<code>clearcolor=<Ein-Aus-Wert></code>	<code>preset: false</code>
--	----------------------------

Sollten beim farbigen Layout die KOMA-Script-Optionen `twoside` sowie auch `open=right` gesetzt sein, so werden beim Aktivieren dieser Option die Rückseiten von Teilen – und je nach Einstellung von [chapterpage](#) gegebenenfalls auch von Kapiteln – farbig gesetzt.³⁶ Die Option wirkt sich ebenfalls auf die Rückseite des Titels aus.³⁷ Der Stil dieser zusätzlich eingefügten Rückseiten ist abhängig von der Option [cleardoublespecialpage](#).

false/off/no:

Es werden weiße Rückseiten bei Titel, Teilen und gegebenenfalls Kapiteln erzeugt.

true/on/yes: (Säumniswert)

Die rückwärtigen Seiten der genannten Elemente des Layouts sind farbig.

2.2.7 DER SATZSPIEGEL

v2.03

<code>cdgeometry=<Einstellung></code>	<code>preset: true</code>
---	---------------------------

Diese Option ist für die Aufteilung beziehungsweise die Berechnung des Satzspiegels verantwortlich. Das Maß der Seitenränder ist im Corporate Design fest vorgegeben und wird standardmäßig von den TUD-KOMA-Script-Klassen eingehalten. Allerdings lassen sich die Seitenränder anpassen, um beispielsweise einen vernünftigen doppelseitigen Satz zu ermöglichen.³⁸ Des Weiteren besteht die Möglichkeit, auf das Standardverhalten von KOMA-Script zurückzufallen und die Satzspiegelberechnung durch das Paket [typearea](#) vornehmen zu lassen. Hier hat insbesondere die Klassenoption `DIV=<Einstellung>` maßgeblichen Einfluss auf den Satzspiegel. Mehr dazu ist in der [Dokumentation von KOMA-Script](#)³⁹ zu finden.

false/off/no:

Die Satzspiegelberechnung erfolgt via [typearea](#), die Vorgaben des Corporate Designs bezüglich der Seitenränder werden ignoriert.

true/on/yes/asymmetric/cd: (Säumniswert)

Die Seitenränder werden im asymmetrischen Stil des Corporate Designs fest definiert und auch für den doppelseitigen Satz (Klassenoption `twoside=true`) genutzt.⁴⁰

symmetric/centred/centered:

Der Satzspiegel wird im einseitigen sowie doppelseitigen Satz auf der Seite zentriert.⁴¹

twoside/balanced:

Diese Einstellung aktiviert den doppelseitigen Satz (`twoside=true`) und verändert den Satzspiegel derart, dass die Ränder der inneren Seiten schmaler sind als die der äußeren.⁴²

Der so erzeugte Satzspiegel ist jedoch nicht sehr vorteilhaft. Es ist zu beachten, dass dabei das Logo der Technischen Universität Dresden sehr nah am inneren Seitenrand des Dokumentes gesetzt wird, folglich insbesondere auf rechten respektive ungeraden Seiten sehr weit an den Blattrand rückt. Diesem Problem kann zumindest teilweise – bei

Achtung!

³⁶Dies führt bei der Ausgabe zu farbigen Blättern (Vorder- und Rückseite) der entsprechenden Elemente des Layouts.

³⁷siehe `\uppertitleback` und `\lowertitleback` im [KOMA-Script-Handbuch](#)

³⁸Hierbei sollte der innere Rand schmaler als der äußere sein

³⁹<http://mirrors.ctan.org/macros/latex/contrib/koma-script/doc/scrguide.pdf>

⁴⁰links: 30 mm, rechts: 20 mm, oben: 25 mm, unten: 30 mm

⁴¹links: 25 mm, rechts: 25 mm, oben: 25 mm, unten: 30 mm

⁴²innen: 20 mm, außen: 30 mm, oben: 25 mm, unten: 30 mm

den Klassen `tudscrbook` und `tudscrartcl` – aus dem Weg gegangen werden, indem Titel, Teile und Kapitel mit der KOMA-Script-Option `open=left` immer auf einer der nächsten linken Seite begonnen werden.

Für die Festlegung der Seitenränder wird das Paket `geometry` genutzt. Ist `cdgeometry=false` gewählt, erfolgt die Berechnung des Satzspiegels durch `typearea`. Die damit berechneten Werte werden anschließend an `geometry` weitergereicht und durch dieses umgesetzt.

v2.03

`extrabottommargin=<Höhe>`

`preset: Opt`

Mit dieser Option kann die Größe des unteren Seitenrandes angepasst werden, wenn der Satzspiegel des Corporate Designs (`cdgeometry=true/symmetric/balanced`) verwendet wird. Insbesondere für den Fall, dass im Fußbereich der Seiten im Stil `tudheadings` entweder mit `\footlogo` Drittlogos verwendet werden und diese über das optionale Argument oder via `\footlogoheight` über die Standardhöhe hinaus vergrößert wurden oder mit `\footcontent` ein übergroßer Inhalt angegeben wurde, kann dieser unter Umständen etwas zu klein sein. Mit der Option `extrabottommargin` wird der Fußbereich durch positive Werte vergrößert, negative Werte verkleinern diesen entsprechend.

Alternativ zu `extrabottommargin` kann auch die Option `cdfoot` mit einer Längenangabe verwendet werden. Dabei spielt es für beide Optionen keine Rolle, ob eine L^AT_EX-Länge, ein T_EX-Abstand oder eine T_EX-Ausdehnung als Länge bei der Wertzuweisung verwendet wird.

KOPF- UND FUSSZEILE IM ZUSAMMENSPIEL MIT DEM SATZSPIEGEL

Da im Corporate Design nicht festgelegt ist, wie die Gestaltung der Kopf- und Fußzeilen in einer wissenschaftlichen Arbeit auszuführen ist, bleibt dem Nutzer dafür eine gewisse Freiheit. Dafür sollte idealerweise das zu KOMA-Script gehörige Paket `scrlayer-scrpage` genutzt werden.

In der Dokumentation zu `typearea` wird auch darauf eingegangen, wann Kopf- und Fußzeile bei der Satzspiegelkonstruktion entweder dem Rand oder dem Textkörper zugeschlagen werden sollten. Dies sollte bei der Erstellung eigener Kopf- und Fußzeilen beachtet werden. Die Einstellung dafür erfolgt mit den beiden KOMA-Script-Optionen `headinclude=<Ein-Aus-Wert>` sowie `footinclude=<Ein-Aus-Wert>`. Diese können – unabhängig von der gewählten Einstellung zur Satzspiegelgestaltung für die Option `cdgeometry` – verwendet werden.

BINDEKORREKTUR

Im Zusammenhang mit Seitenrändern und dem Satzspiegel ist die durch das Paket `typearea` angebotene KOMA-Script-Option `BCOR=<Länge>` zu erwähnen. Mit dieser kann bei der Satzspiegelberechnung ein Heftrand beziehungsweise eine Bindekorrektur berücksichtigt werden. Der mit dieser Option angegebene Wert wird durch die TUD-KOMA-Script-Klassen auch an das Paket `geometry` weitergereicht, so dass der Benutzer unabhängig von der Satzspiegelgestaltung (`cdgeometry`) die Option `BCOR` nutzen kann. So lässt sich eine Bindekorrektur von beispielsweise 5 mm mit der *Klassenoption* `BCOR=5mm` festlegen.

Eine Anpassung der Bindekorrektur hat natürlich *immer* eine Änderung der verfügbaren Breite des Textbereichs zur Folge hat und führt somit zwingend zu einer Anpassung des Satzspiegels. Da die Bindekorrektur jedoch abhängig von der Höhe des Buchblocks gewählt werden sollte, welche letztendlich erst mit dem Druck des fertiggestellten Dokumentes bestimmt werden kann, muss diese zu Beginn abgeschätzt werden.

Beispiel: Als Faustregel gilt, dass die erforderliche Bindekorrektur in etwa der halben Höhe des Buchblocks entsprechen sollte. Dessen Höhe wiederum ist abhängig von der Anzahl der Seiten sowie der Dichte des verwendeten Papiers. Wird normales Papier mit

einer Dichte von 80 g/m² verwendet, so entsprechen 100 Blatt in etwa einer Höhe von 10 mm, bei 100 g/m² ca. 12 mm. Dementsprechend wäre eine Bindekorrektur von BCOR=5mm beziehungsweise BCOR=6mm bei diesem Beispiel zu wählen.

2.2.8 VERWENDUNG VON SCHRIFTELEMENTEN

Von TUD-KOMA-Script werden weitere Schriftelemente – in Ergänzung zu den bereits durch KOMA-Script bereitgestellten – definiert. Dies sind `titlepage`, `thesis`, `tudheadings` sowie `parttitle`. Sowohl die bereits durch KOMA-Script definierten als auch alle hier genannten Schriftelemente und später erläuterten sollten im Bedarfsfall durch den Anwender über den Befehl `\addtokomafont{<Schriftelement>}{<Einstellungen>}` angepasst werden. Mehr dazu ist im [KOMA-Script-Handbuch](#)⁴³ innerhalb des Abschnitts *Textauszeichnungen* zu finden.

2.2.9 DIE FARBEN DES CORPORATE DESIGNS

Zur Verwendung der Farben des Corporate Designs wird das Paket `tudscrcolor` genutzt. Falls dieses nicht in der Präambel geladen wird – um beispielsweise zusätzliche Optionen aufzurufen – binden die TUD-KOMA-Script-Klassen dieses automatisch ein. Detaillierte Informationen sind in der Dokumentation von `tudscrcolor` in [Unterkapitel 4.2 auf Seite 63](#) zu finden.

2.3 ZUSÄTZLICHE OPTIONEN UND ERWEITERUNGEN

Neben den Befehlen für die Anpassung des Layouts an das Corporate Design der Technischen Universität Dresden stellen die TUD-KOMA-Script-Klassen weitere Befehle und Umgebungen zur Verfügung, um die Anwendung insbesondere für wissenschaftliche Arbeiten zu erleichtern.

2.3.1 ZUSAMMENFASSUNG/KURZFASSUNG

v2.02
v2.04

```
abstract=<Einstellung>
```

Diese Option wird bereits durch KOMA-Script für die Klassen `scrartcl` und `scrreprt` standardmäßig bereitgestellt. Für die Klasse `scrbook` geschieht dies nicht. Dazu heißt es im Handbuch:

Bei Büchern wird in der Regel eine andere Art der Zusammenfassung verwendet. Dort setzt man ein entsprechendes Kapitel an den Anfang oder Ende des Werks. Oft wird diese Zusammenfassung entweder mit der Einleitung oder einem weiteren Ausblick verknüpft. Daher gibt es bei `scrbook` generell keine `abstract`-Umgebung. Bei Berichten im weiteren Sinne, etwa einer Studien- oder Diplomarbeit, ist ebenfalls eine Zusammenfassung in dieser Form zu empfehlen.

Durch die TUD-KOMA-Script-Klassen wird die `abstract`-Option erweitert. Neben den Auswahlmöglichkeit, welche bereits KOMA-Script für die Klassen `tudscrartcl` und `tudscrreprt` anbietet, kann die Überschrift für die Zusammenfassung außerdem in Gestalt eines Unterkapitels oder für `tudscrreprt` und `tudscrbook` in der Form eines Kapitels ausgegeben werden.

`false/off/no`: nur für `tudscrartcl` und `tudscrreprt` verfügbar

Es wird keine Überschrift für die `abstract`-Umgebung ausgegeben.

⁴³<http://mirrors.ctan.org/macros/latex/contrib/koma-script/doc/scrguide.pdf>

`true/on/yes`: (Säumniswert) nur für `tudscrartcl` und `tudscrreprt` verfügbar

Wie bei den KOMA-Script-Klassen wird eine zentrierte Überschrift mit dem Bezeichner `\abstractname` vor der eigentlichen Zusammenfassung gesetzt.

`section/addsec`:

Die Überschrift (`\abstractname`) verwendet den Gliederungsbefehl `\section`.

`chapter/addchap`: (Säumniswert für `tudscrbook`) nur für `tudscrreprt` und `tudscrbook` verfügbar

Es wird der Befehl `\chapter` für das Setzen der Überschrift (`\abstractname`) genutzt.

`heading`:

Es wird die höchstmögliche Gliederungsebene verwendet. Für `tudscrartcl` entspricht dies `abstract=section`, bei `tudscrreprt` und `tudscrbook` `abstract=chapter`.

Abhängig von der gewählten Gliederungsebene der Überschrift wird das Verhalten für das Setzen eines Eintrages ins Inhaltsverzeichnis festgelegt. Ohne oder mit zentrierter Überschrift wird per Voreinstellung kein Eintrag erzeugt. Wird die Überschrift jedoch in Form einer Gliederungsebene gewählt, so erscheint die Zusammenfassung für gewöhnlich im Inhaltsverzeichnis auf der obersten Ebene. Das voreingestellte Verhalten für die Einträge ins Inhaltsverzeichnis kann jederzeit mit folgenden Werten durch den Anwender überschrieben werden.

`notoc/nottotoc`:

Die Zusammenfassung wird definitiv nicht ins Inhaltsverzeichnis eingetragen.

`toc/totoc`:

Es wird ein nicht nummerierter Eintrag im Inhaltsverzeichnis auf der obersten Gliederungsebene der verwendeten Dokumentklasse für die Zusammenfassung gesetzt.

v2.02

`leveldown/tocleveldown/totocleveldown`:

Der Inhaltsverzeichniseintrag wird eine Gliederungsebene unterhalb der obersten erzeugt.

v2.04

`tocmultiple/totocmultiple/tocaggregate/totocaggregate`:

Es wird ein *einziger* Inhaltsverzeichniseintrag für *alle* Zusammenfassungen erstellt.

v2.02

Außerdem kann das Verhalten für die Kolumnentitel durch den Nutzer beeinflusst werden. Diese werden normalerweise automatisch gesetzt, wenn diese über die KOMA-Script-Option `automark` aktiviert wurden und sind von der aktuellen Gliederungsebene der Überschrift abhängig. Werden jedoch mit Hilfe der KOMA-Script-Option `manualmark` manuelle Kolumnentitel genutzt, müssen diese normalerweise auch für die Zusammenfassung durch den Anwender gesetzt werden. Mit `abstract=markboth` lässt sich allerdings das Setzen der Kolumnentitel unabhängig davon forcieren.

`markboth`:

Unabhängig von der Verwendung manueller oder automatischer Kolumnentitel werden diese auf rechten sowie linken Seiten mit `\abstractname` gesetzt.

`nomarkboth`:

Die Einstellung für manuelle oder automatische Kolumnentitel werden beachtet und abhängig von der verwendeten Gliederungsebene der Überschrift gesetzt.

Mit dem optionalen Parameter `markboth` der `abstract`-Umgebung kann der Kolumnentitel mit einem beliebigen Inhalt gesetzt werden.

Häufig wird für Abschlussarbeiten verlangt, neben der deutschsprachigen auch noch eine englischsprachige Zusammenfassung zu verfassen. Mit der Einstellung `abstract=multiple` lassen sich mehrere Zusammenfassungen auf einer Seite ausgeben – sofern genügend Platz vorhanden ist. Außerdem kann die standardmäßige vertikale Zentrierung der `abstract`-Umgebung auf einer Seite unterdrückt werden. Diese Einstellungen zur Positionierung der Zusammenfassungen innerhalb der `abstract`-Umgebung werden nur wirksam, wenn eine Titel-

seite (KOMA-Script-Option `titlepage=true`) und *keine* Überschriften in Form von Kapiteln (`abstract=chapter`) verwendet werden.

`single/one/simple`:

Jede Zusammenfassung wird auf einer eigenen Seite beziehungsweise im zweispaltigen Satz in einer neuen Spalte ausgegeben.

v2.02

`multiple/multi/all/aggregate`:

Zusammenfassungen, welche mit `\nextabstract` getrennt wurden, werden direkt nacheinander auf der gleichen Seite ausgegeben, wenn ausreichend Platz auf dieser vorhanden sein sollte. Ist die Klassenoption `twocolumn` aktiviert, erfolgt die Ausgabe aller Zusammenfassungen ohne Spaltenumbruch.

`fil/fill/vfil/vfill`:

Alle Zusammenfassungen auf einer Ausgabeseite werden vertikal zentriert. Für den zweispaltigen Satz (Klassenoption `twocolumn`) steht diese Einstellung nicht zur Verfügung.

`nofil/nofill/novfil/novfill`:

Die Ausgabe erfolgt wie im normalen Fließtext auch.

v2.02

```
\begin{abstract}[<Sprache> || <Parameterliste>]
...
\end{abstract}
\nextabstract[<Sprache> || <Parameterliste>]
language=<Sprache> (Parameter)
markboth=<Ein-Aus-Wert> || <Kolumnentitel> (Parameter)
pagestyle=<Seitenstil> (Parameter)
columns=<Anzahl> (Parameter)
option=<Einstellung> (Parameter)
```

(siehe `abstract` auf Seite 40)

Die `abstract`-Umgebung dient speziell für die Ausgabe einer Zusammenfassung, entweder zu Beginn eines Dokumentes oder beispielsweise vor einem Teil oder Kapitel. Wird ein Titelkopf und keine Titelseite verwendet (KOMA-Script-Option `titlepage=false`), wird eine Zusammenfassung, die *nicht* mit der Überschrift einer Gliederungsebene gesetzt wird, wie bei den KOMA-Script-Klassen in einer `quotation`-Umgebung gesetzt, um diese vom restlichen Fließtext abzuheben. Diese hat jedoch den Nachteil, dass in besagter Umgebung die KOMA-Script-Option `parskip` nicht beachtet wird. Um dieses Problem zu beheben, kann das Paket `quoting` geladen werden, wodurch stattdessen die Umgebung `quoting` verwendet wird.

Mit der zuvor erläuterten Option `abstract` kann eingestellt werden, in welcher Gestalt die Zusammenfassung ausgegeben werden soll. Des Weiteren lässt sich jede `abstract`-Umgebung individuell über weitere Parameter als optionales Argument anpassen. Damit lassen sich gegebenenfalls für eine bestimmte `abstract`-Umgebung die globalen Einstellungen der Option `abstract` lokal ändern und gezielt anpassen.

Wird das Paket `babel` durch den Anwender geladen, kann mit dem optionalen Parameter `language=<Sprache>` die Sprache innerhalb der `abstract`-Umgebung geändert werden. Dafür muss die gewünschte Sprache bereits mit dem Laden von `babel` entweder als Paketoption oder besser noch als Klassenoption angegeben worden sein. Dadurch werden innerhalb der Umgebung die Bezeichnung `\abstractname` und die Trennungsmuster sprachspezifisch angepasst. Die gewünschte Sprache kann auch ohne die Verwendung des Parameters `language` direkt als optionales Argument übergeben werden.

v2.02

Mit `markboth` können die gesetzten Kolumnentitel beeinflusst werden. Wird `markboth=false`

angegeben, werden automatische respektive manuelle Kolumnentitel verwendet. Die Einstellung `markboth=true` wiederum setzt diese für linke und rechte Seiten auf `\abstractname`. Außerdem lässt sich der Kolumnentitel mit `markboth=<Kolumnentitel>` auch direkt festlegen. So können die Kolumnen beispielsweise mit der Verwendung von `markboth={}` auch gelöscht werden. Sollte `markboth` aktiviert werden, so wird in der Umgebung automatisch der Seitenstil `headings` genutzt – falls eine Titelseite und kein Titelkopf (KOMA-Script-Option `titlepage=true`) verwendet wird. Mit dem Parameter `pagestyle` kann dieser auch manuell angegeben werden, die `tudheadings`-Seitenstile werden dabei ebenfalls unterstützt werden.

Wurde das Paket `multicol` geladen, kann mit dem Parameter `columns=<Anzahl>` die Zusammenfassung mehrspaltig gesetzt werden. Dem Parameter `option` können alle gültigen, bereits erläuterten Werte der Option `abstract` übergeben werden. Die damit gemachten Einstellungen wirken sich – im Gegensatz zur Angabe als Klassenoption oder über die Variante der späten Optionenwahl⁴⁴ – lediglich lokal auf die verwendete `abstract`-Umgebung aus.

v2.02

Sollen mehrere Zusammenfassungen im gleichen Stil erzeugt und die Einstellungen der Option `abstract=simple/multiple/fill/nofill` beachtet werden, so ist die `abstract`-Umgebung nur einmal zu verwenden. Innerhalb dieser müssen die einzelnen Zusammenfassungen mit `\nextabstract` voneinander getrennt werden. Der Befehl akzeptiert dabei im optionalen Argument alle Parameter, die auch von der `abstract`-Umgebung selbst unterstützt werden. Das Minimalbeispiel in [Unterkapitel 5.2 auf Seite 75](#) zeigt hierfür das notwendige Vorgehen.

Wird die `abstract`-Umgebung innerhalb des Argumentes der Befehle `\setpartpreamble` beziehungsweise `\setchapterpreamble` verwendet, so wird die Überschrift – im Fall, dass nicht `abstract=false` gewählt ist – *immer* in Textgröße und zentriert gesetzt.

UMBENENNUNG DER ZUSAMMENFASSUNG

Mit dem KOMA-Script-Befehl `\renewcaptionname` kann der Bezeichner – sprich der Wortlaut – der für die `abstract`-Umgebung verwendeten Überschrift verändert werden. Mehr dazu ist in [Unterkapitel 2.4](#) zu finden.

Beispiel: Die Überschrift der `abstract`-Umgebung soll für die Sprache `ngerman` von „Zusammenfassung“ in „Kurzfassung“ umbenannt werden. Der Befehl `\renewcaptionname` erwartet die drei obligatorischen Argumente `{<Sprache>}{<Makro>}{<Inhalt>}`:

```
\renewcaptionname{ngerman}{\abstractname}{Kurzfassung}
```

2.3.2 SELBSTSTÄNDIGKEITSERKLÄRUNG UND SPERRVERMERK

v2.02
v2.04

```
declaration=<Einstellung>      preset: true
```

Mit `declaration` kann äquivalent zur Option `abstract` die Gestaltung von Selbstständigkeitsklärung und Sperrvermerk angepasst werden. Zur Ausgabe der Erklärungen werden die Umgebung `declarations` sowie die Befehle `\declaration` beziehungsweise `\confirmation` und `\blocking` bereitgestellt.

Die beiden Optionen `abstract` und `declaration` ähneln sich sehr stark. Alle möglichen Wertzuweisungen für `declaration` wurden bereits bei der Beschreibung von `abstract` ausführlich erläutert. Deshalb geschieht dies hier in einer etwas kürzeren Ausführung. Sollte Ihnen eine Erläuterung etwas dürrig erscheinen, so hilft mit Sicherheit ein Blick zur Erklärung der Option `abstract` in [Abschnitt 2.3.1 auf Seite 40](#).

⁴⁴`\TUDoption{abstract}{<Einstellung>}` oder `\TUDoptions{abstract=<Einstellung>}`

Die möglichen Werte für die Gestaltung der Überschrift werden nachfolgend genannt. Im Gegensatz zur Option `abstract` stehen die Einstellungen `declaration=true/false` auch für die Klasse `tudscrbook` zur Verfügung.

`false/off/no`:

Es wird keine Überschrift über den Erklärungen selbst ausgegeben.

`true/on/yes`: (Säumniswert)

Eine zentrierte Überschrift mit dem Bezeichner `\confirmationname` vor der Selbstständigkeitserklärung beziehungsweise `\blockingname` vor dem Sperrvermerk wird gesetzt.

`section/addsec`:

Die Überschrift verwendet den Gliederungsbefehl `\section`.

`chapter/addchap`: (Säumniswert für `tudscrbook`) nur für `tudscrreprt` und `tudscrbook` verfügbar

Es wird der Befehl `\chapter` für das Setzen der Überschrift genutzt.

`heading`:

Es wird die höchstmögliche Gliederungsebene verwendet. Für `tudscrartcl` entspricht dies `declaration=section`, bei `tudscrreprt` und `tudscrbook` `declaration=chapter`.

Abhängig von der gewählten Gliederungsebene der Überschrift wird das Verhalten für das Setzen eines Eintrages ins Inhaltsverzeichnis festgelegt. Normalerweise wird nur für Überschriften in Form einer Gliederungsebene ein Eintrag der Erklärung ins Inhaltsverzeichnis erstellt, für `declaration=true/false` geschieht dies standardmäßig nicht. Das voreingestellte Verhalten kann mit folgenden Werten überschrieben werden.

`notoc/nottotoc`:

Die Erklärung wird definitiv nicht ins Inhaltsverzeichnis eingetragen.

`toc/totoc`:

Unabhängig von der Wahl der Überschrift erhält jede Erklärung einen nicht nummerierten Eintrag im Inhaltsverzeichnis auf der obersten Gliederungsebene der aktuell gerade verwendeten Dokumentklasse.

v2.02

`leveldown/tocleveldown/totocleveldown`:

Der Inhaltsverzeichniseintrag wird eine Gliederungsebene unterhalb der obersten erzeugt.

v2.04

`tocmultiple/totocmultiple/tocaggregate/totocaggregate`:

Es wird ein *einzig*er Inhaltsverzeichniseintrag für *alle* Erklärungen erstellt.

v2.02

Normalerweise werden die automatischen Kolumnentitel in Abhängigkeit von der Gliederungsebene der Überschrift gesetzt, falls diese denn aktiviert sind über die KOMA-Script-Option `automark`. Werden manuelle Kolumnentitel genutzt, müssen diese auch für die Erklärungen manuell gesetzt werden. Mit `declaration=markboth` lässt sich unabhängig davon das Setzen der Kolumnentitel auf linken und rechten Seiten forcieren, wobei hierfür der Titel der Überschrift genutzt wird.

`markboth`:

Unabhängig von der Verwendung manueller oder automatischer Kolumnentitel werden diese auf rechten sowie linken Seiten mit den Bezeichnern `\confirmationname` beziehungsweise `\blockingname` gesetzt.

`nomarkboth`:

Die Einstellung für manuelle oder automatische Kolumnentitel werden beachtet.

Für `\declaration` respektive `\confirmation` und `\blocking` sowie die `declaration`-Umgebung lässt sich mit dem Parameter `markboth` ein beliebiger Kolumnentitel setzen.

Die folgenden Einstellungen zur Positionierung der Erklärungen haben lediglich Auswirkungen, wenn die Überschrift der Erklärung *nicht* im Form eines Kapitels ausgegeben und mit der

KOMA-Script-Option `titlepage=true` eine Titelseite verwendet wird.

`single/one/simple`:

Jede Erklärung wird auf einer separaten Seite beziehungsweise im zweispaltigen Satz in einer neuen Spalte ausgegeben.

v2.02

`multiple/multi/all/aggregate`:

Erklärungen, welche in der `declarations`-Umgebung mit den Befehlen `\confirmation`, `\blocking` und `\declaration` oder außerhalb dieser mit `\declaration` gesetzt wurden, werden direkt nacheinander auf der gleichen Seite ausgegeben, wenn ausreichend Platz auf dieser vorhanden sein sollte. Ist die Klassenoption `twocolumn` aktiviert, erfolgt die Ausgabe aller Erklärungen ohne Spaltenumbruch.

`fil/fill/vfil/vfill`:

Alle Erklärungen auf einer Ausgabeseite werden vertikal zentriert. Für den zweispaltigen Satz (Klassenoption `twocolumn`) steht diese Einstellung nicht zur Verfügung.

`nofil/nofill/novfil/novfill`:

Die Ausgabe erfolgt wie im normalen Fließtext auch.

v2.02

v2.04

```
\begin{declarations}[<Sprache> || <Parameterliste>]
...
\end{declarations}
\nextdeclaration[<Sprache> || <Parameterliste>]{<Überschrift>}{<Erklärung>}
language=<Sprache> (Parameter)
markboth=<Ein-Aus-Wert> || <Kolumnentitel> (Parameter)
pagestyle=<Seitenstil> (Parameter)
columns=<Anzahl> (Parameter)
option=<Einstellung> (Parameter)
supporter=<Unterstützer> (Parameter)
place=<Ort> (Parameter)
closing=<Ende> (Parameter)
company=<Firma> (Parameter)
```

Für Selbstständigkeitserklärung und Sperrvermerk sollten im einfachsten Fall die Befehle `\declaration` beziehungsweise `\confirmation` und `\blocking` verwendet werden. Sobald diese jedoch in anderer Reihenfolge, mehrfacher Ausführung, unterschiedlichen Sprachen oder durch zusätzliche Erklärungen ergänzt werden, so bietet die `declarations`-Umgebung die notwendigen Freiheiten.

Innerhalb dieser Umgebung können Selbstständigkeitserklärung und Sperrvermerk mit dem Befehl `\declaration` direkt nacheinander folgend beziehungsweise mit `\confirmation` und `\blocking` auch separat ausgegeben werden. Dies kann in beliebiger Reihenfolge und auch mehrmals geschehen, um diese beispielsweise mehrsprachig zu setzen. Des Weiteren gibt es mit `\nextdeclaration` die Möglichkeit, eine Erklärung völlig frei zu verfassen. Dieser Befehl kann *ausschließlich* innerhalb der `declarations`-Umgebung genutzt werden, wobei im ersten Argument die gewünschte Überschrift und im zweiten der Inhalt respektive Text der Erklärung selbst angegeben werden muss.

v2.04

Die im Folgenden beschriebenen Parameter können sowohl für die Umgebung `declarations` selbst als auch für die zuvor genannten Befehle als optionales Argument verwendet werden. Ähnlich wie die gleichnamigen Optionen sind auch die Umgebungen `abstract` und `declaration` sehr ähnlich zueinander. Deshalb werden die Erläuterungen relativ kurz gehalten. Ist ein Erklä-

rung für einen Parameter etwas unverständlich, kann diese bei der Umgebung `abstract` in [Abschnitt 2.3.1 auf Seite 42](#) nachgelesen werden.

Wurde das Paket `babel` geladen, kann die Sprache – sofern diese als Paketoption oder besser noch als Klassenoption angegeben wurde – mit dem Parameter `language=<Sprache>` für die `declarations`-Umgebung geändert werden. Dadurch werden die Bezeichner – unter anderem `\confirmationname` und `\blockingname` – sowie die Trennungsmuster innerhalb der Umgebung sprachspezifisch angepasst.

v2.02

Die Kolumnentitel können mit `markboth` beeinflusst werden. Mit `markboth=true` werden für diese auf linker und rechter Seite `\confirmationname` respektive `\blockingname` verwendet. Außerdem kann der Anwender selbige mit `markboth=<Kolumnentitel>` auch direkt festlegen. Sollte `markboth` verwendet werden, wird der Seitenstil automatisch auf `headings` gesetzt. Mit dem Parameter `pagestyle` lässt sich dieser für die Umgebung auch manuell angeben. Wurde das Paket `multicol` geladen, wird mit `columns=<Anzahl>` der Inhalt der Umgebung mehrspaltig gesetzt. Für `option` können alle gültigen Werte der Option `declaration` angegeben werden. Die Verwendung der Parameter `supporter` sowie `place` und `closing` ist in der Dokumentation des Befehls `\confirmation` zu finden, der Parameter `company` ist für `\blocking` erläutert.

v2.02

```
\confirmation[<Unterstützer> || <Parameterliste>]
supporter=<Unterstützer> (Parameter)
place=<Ort> (Parameter)
closing=<Ende> (Parameter)
language=<Sprache> (Parameter)
markboth=<Ein-Aus-Wert> || <Kolumnentitel> (Parameter)
pagestyle=<Seitenstil> (Parameter)
columns=<Anzahl> (Parameter)
option=<Einstellung> (Parameter)
```

Mit diesem Befehl wird ein sprachspezifischer Standardtext für eine Selbstständigkeitserklärung ausgegeben, welcher in `\confirmationtext` gespeichert ist. Wie dieser angepasst beziehungsweise geändert werden kann, ist unter [Unterkapitel 2.4](#) zu finden. Er kann sowohl innerhalb der `declarations`-Umgebung als auch außerhalb dieser direkt im Dokument verwendet werden.

Wird `\confirmationtext` nicht geändert, kann dieser über das optionale Argument von `\confirmation` und die deklarierten Parameter angepasst werden. Im Standardtext der Selbstständigkeitserklärung werden sowohl der Titel als auch der Typ der Abschlussarbeit – falls dieser mit `\thesis`, `\subject{<Tabelle 2.1>}` beziehungsweise mit der Option `subjectthesis` angegeben wurde – aufgeführt. Über den Parameter `supporter` oder zuvor mit dem Befehl `\supporter` können weitere an der Arbeit beteiligte Personen angegeben werden. Mehrere zu nennende Personen sind auch hier durch `\and` zu trennen. Das Feld der Unterstützer kann auch mit dem bloßen optionalen Argument ohne die Angabe eines Parameters angepasst werden.

Nach dem eigentlichen Text der Selbstständigkeitserklärung wird der mit `place` beziehungsweise `\place` angegebene Ort sowie das mit `\date` eingestellte Datum ausgegeben. Als Voreinstellung ist für den Ort „Dresden“ gewählt. Danach folgen – mit etwas vertikalem Freiraum für die notwendige Unterschrift – der Autor oder die Autoren, angegeben durch den Befehl `\author`. Soll anstelle dessen etwas anderes nach dem Text der Selbstständigkeitserklärung gesetzt werden, kann dies mit dem Parameter `closing` oder zuvor mit dem Befehl `\confirmationclosing` angepasst werden. Die Parameter `language`, `markboth`, `pagestyle`, `columns` und `option` entsprechen in ihrem Verhalten denen der `declarations`-Umgebung.


```
\blocking[<Firma> || <Parameterliste>]
company=<Firma> (Parameter)
language=<Sprache> (Parameter)
markboth=<Ein-Aus-Wert> || <Kolumnentitel> (Parameter)
pagestyle=<Seitenstil> (Parameter)
columns=<Anzahl> (Parameter)
option=<Einstellung> (Parameter)
```

Beim Sperrvermerk verhält es sich äquivalent zur Selbstständigkeitserklärung. Es wird der in `\blockingtext` hinterlegte Standardtext in der gewählten Sprache ausgegeben. Dieser kann durch den Anwender geändert werden. Wie genau ist in [Unterkapitel 2.4](#) beschrieben. Der Befehl `\blocking` kann sowohl innerhalb der Umgebung `declarations` als auch außerhalb direkt im Dokument verwendet werden.

In seiner ursprünglichen Definition, kann er im optionalen Argument über die deklarierten Parameter angepasst werden. Im Standardtext des Sperrvermerks werden sowohl der Titel als auch der Typ der Abschlussarbeit – falls dieser mit `\thesis`, `\subject{<Tabelle 2.1>}` respektive mit der Option `subjectthesis` angegeben wurde – aufgeführt. Mit `company` oder vorher mit `\company` kann zusätzlich eine im Sperrvermerk zu nennende Firma oder ähnliches angegeben werden. Dieses Feld kann auch direkt im optionalen Argument ohne die Verwendung eines Parameters gesetzt werden. Die weiteren Parameter `language`, `markboth`, `pagestyle`, `columns` und `option` entsprechen in ihrem Verhalten denen der `declarations`-Umgebung.

```
\declaration[<Parameterliste>]
language=<Sprache> (Parameter)
markboth=<Ein-Aus-Wert> || <Kolumnentitel> (Parameter)
pagestyle=<Seitenstil> (Parameter)
columns=<Anzahl> (Parameter)
option=<Einstellung> (Parameter)
supporter=<Unterstützer> (Parameter)
place=<Ort> (Parameter)
closing=<Ende> (Parameter)
company=<Firma> (Parameter)
```

Dieser Befehl gibt die Selbstständigkeitserklärung und den Sperrvermerk direkt aufeinanderfolgend aus. Dabei werden die Einstellungen zur Positionierung der einzelnen Erklärungen, welche über die Wertzuweisungen an die Option `declaration=simple/multiple/fill/nofill` erfolgen, beachtet. Er kann sowohl innerhalb der `declarations`-Umgebung als auch außerhalb direkt im Dokument verwendet werden und akzeptiert im optionalen Argument dabei alle für die `declarations`-Umgebung beschriebenen Parameter.

```
\supporter{<Unterstützer>}
\place{<Ort>}
\confirmationclosing{<Ende>}
\company{<Firma>}
```

Diese Makros ändern – im Gegensatz zu den Parametern der bereits vorgestellten Befehle `\confirmation` und `\blocking` – die entsprechenden Feldwerte für das gesamte Dokument. Genutzt werden kann dies beispielsweise wenn ein Erklärungstyp in unterschiedlichen Sprachen ausgegeben wird. Hiermit kann man sich die mehrfache Angabe eines Parameters sparen.

2.3.3 FUSSNOTEN IN ÜBERSCHRIFTEN

v2.02

```
footnotes=<Einstellung>  
symbolheadings (Zähler)
```

```
preset: nosymbolheadings
```

Für die Überschriften wird die KOMA-Script-Option `footnotes` erweitert. Normalerweise kann diese die Werte `multiple` und `nomultiple` annehmen, wobei Letzteres der Standardfall ist. Die TUD-KOMA-Script-Hauptklassen erweitern die Option dahingehend, dass auf die Verwendung von Symbolen anstelle von Zahlen innerhalb der Überschriften umgeschaltet werden kann. Hierfür wird der Zähler `symbolheadings` definiert, der mit dem Beginn eines neuen Kapitels zurückgesetzt wird.

`nosymbolheadings/numberheadings`:

Die Fußnoten der Überschriften werden fortlaufend mit denen des Fließtextes gesetzt.

`symbolheadings`:

Für die Überschriften werden symbolische Fußnoten mit einem eigenen Zähler verwendet.

2.3.4 LESEZEICHEN

```
tudbookmarks=<Ein-Aus-Wert>
```

```
preset: true
```

Diese Option wird wirksam, wenn `hyperref` geladen wurde. Es werden für die Umschlag- und Titelseite, das Inhaltsverzeichnis sowie – bei der Verwendung des Paketes `tudscrsupervisor` – die Aufgabenstellung Lesezeichen oder auch Outline-Einträge im PDF-Dokument erzeugt.

`false/off/no`:

Es erfolgt kein Eintrag von ergänzenden Lesezeichen.

`true/on/yes`: (Säumniswert)

Es werden automatisch zusätzliche Lesezeichen eingetragen.

```
\tudbookmark[<Ebene>]{<Text>}{<Ankername>}
```

Der Befehl `\tudbookmark` arbeitet wie `\pdfbookmark` aus `hyperref` mit dem Unterschied, dass die Lesezeichen nur generiert werden, wenn die Option `tudbookmarks` aktiviert ist.

2.4 SPRACHABHÄNGIGE BEZEICHNER

Durch KOMA-Script werden Befehle, mit denen sprachabhängige Bezeichner erzeugt oder geändert werden können, zur Verfügung gestellt. Diese werden durch TUD-KOMA-Script genutzt, um lokalisierte Begriffe für die Sprachen Englisch und Deutsch bereitzustellen. Ein Großteil davon betrifft Bezeichnungen für Felder auf der Titelseite ([Abschnitt 2.2.3](#)). Hierfür wird `\providecaptionname{<Sprache>}{<Makro>}{<Inhalt>}` verwendet, wobei `<Sprache>` dem geladenen Sprachpaket – normalerweise das Paket `babel` – bekannt sein muss.

Sollte der Anwender die im Folgenden erläuterten oder auch andere Bezeichner, welche von einem beliebigen (Sprach-)Paket bereitgestellt werden, ändern wollen, ist hierfür der Befehl `\renewcaptionname{<Sprache>}{<Makro>}{<Inhalt>}` zu verwenden. Es sollte natürlich dabei eine `<Sprache>` angegeben werden, welche im Dokument durch `babel` oder ein anderes Sprachpaket verwendet wird, beispielsweise `ngerman` oder `english`.

Die Makros der Bezeichner und deren Verwendung werden folgend kurz beschrieben und tabellarisch aufgeführt. Dabei wurde versucht, alle Befehle der Bezeichner für bestimmte

Begriffe auf ...name und beschreibende Texte auf ...text enden zu lassen.

v2.02

```
\supervisorname (Bezeichner)
\supervisorothername (Bezeichner)
\refereename (Bezeichner)
\refereeothername (Bezeichner)
\advisorname (Bezeichner)
\advisorothername (Bezeichner)
\professorname (Bezeichner)
\professorothername (Bezeichner)
```

Diese sprachabhängigen Begriffe sind die Bezeichner für die Titelseitenfelder von Betreuer ([\supervisor](#)), Gutachter ([\referee](#)) und Fachreferent ([\advisor](#)). Soll innerhalb eines dieser Felder mehr als eine Person angegeben werden, so sind die Einzelpersonen jeweils mit dem Befehl [\and](#) voneinander zu trennen. In diesem Fall werden alle nach der erstgenannten folgenden Personen durch den Bezeichner `\...othername` ergänzt.

v2.02

Bei der Bezeichnung des Gutachters wird unterschieden, ob einer oder mehrere angegeben wurden. Wird lediglich einer genannt, so ist eine Unterscheidung nicht notwendig. Werden jedoch zwei Gutachter angegeben, so werden diese auch mit Erst- und Zweitgutachter betitelt. Für den betreuenden Hochschullehrer ([\professor](#)) wird ähnlich verfahren. Hier wird allerdings lediglich die Bezeichnung vom Singular in den Plural gegebenenfalls automatisch geändert.

Bezeichner	Deutsch	Englisch
\supervisorname	Betreuer	Supervisor
\supervisorothername	<leer>	<empty>
\refereename	Gutachter/Erstgutachter	Referee/First referee
\refereeothername	Zweitgutachter	Second referee
\advisorname	Fachreferent	Advisor
\advisorothername	<leer>	<empty>
\professorname	Betreuende(r) Hochschullehrer	Supervising professor(s)
\professorothername	<leer>	<empty>

```
\dissertationname (Bezeichner)
\diplomathesisname (Bezeichner)
\masterthesisname (Bezeichner)
\bachelorthesisname (Bezeichner)
\studentresearchname (Bezeichner)
\projectpapername (Bezeichner)
\seminarpapername (Bezeichner)
\researchname (Bezeichner)
\logname (Bezeichner)
\internshipname (Bezeichner)
\reportname (Bezeichner)
```

Diese Bezeichner dienen zur Typisierung speziell für eine Abschlussarbeit. Wie diese genutzt werden können, ist bei der Erläuterung von [\thesis](#) und [\subject](#) in [Abschnitt 2.2.3 auf Seite 33](#) beziehungsweise in [Tabelle 2.1](#) zu finden.

Bezeichner	Deutsch	Englisch
<code>\dissertationname</code>	Dissertation	Dissertation
<code>\diplomathesisname</code>	Diplomarbeit	Diploma Thesis
<code>\masterthesisname</code>	Master-Arbeit	Master Thesis
<code>\bachelorthesisname</code>	Bachelor-Arbeit	Bachelor Thesis
<code>\studentresearchname</code>	Studienarbeit	Student Research Project
<code>\projectpapername</code>	Projektarbeit	Project Paper
<code>\seminarpapername</code>	Seminararbeit	Seminar Paper
<code>\researchname</code>	Forschungsbericht	Research Report
<code>\logname</code>	Protokoll	Log
<code>\internshipname</code>	Praktikumsbericht	Internship Report
<code>\reportname</code>	Bericht	Report

`\dateofbirthtext` (Bezeichner)
`\placeofbirthtext` (Bezeichner)
`\matriculationnumbername` (Bezeichner)
`\matriculationyearname` (Bezeichner)

Werden für den Autor oder die Autoren das Geburtsdatum (`\dateofbirth`), der Geburtsort (`\placeofbirth`) sowie die Matrikelnummer (`\matriculationnumber`) und/oder das Immatrikulationsjahr (`\matriculationyear`) angegeben, werden sowohl auf der Titelseite als auch auf der gegebenenfalls mit **tudscr**supervisor erstellten Aufgabenstellung die dazugehörigen Bezeichner vorangestellt. Auf dem Titel werden diese dabei mit dem durch `\titledelimiter` gegebenen Trennzeichen vom eigentlichen Feld abgegrenzt.

Bezeichner	Deutsch	Englisch
<code>\dateofbirthtext</code>	Geboren am	Born on
<code>\placeofbirthtext</code>	in	in
<code>\matriculationnumbername</code>	Matrikelnummer	Matriculation number
<code>\matriculationyearname</code>	Immatrikulationsjahr	Matriculation year

v2.02

`\graduationtext` (Bezeichner)

Wurde erkannt, dass das Dokument eine Abschlussarbeit ist,⁴⁵ so kann der zu erlangende akademische Grad mit dem Befehl `\graduation` angegeben werden. Bei dessen Ausgabe auf dem Titel wird dabei der entsprechende Text dazu angegeben.

Bezeichner	Deutsch	Englisch
<code>\graduationtext</code>	zur Erlangung des akademischen Grades	to achieve the academic degree

`\datetext` (Bezeichner)
`\defensedatetext` (Bezeichner)

Wird mit `\date` das Datum und mit `\defensedate` ein Datum der Verteidigung für eine Abschlussarbeit angegeben, so werden auch diese Felder durch einen einleitenden Text beschrieben.

⁴⁵Entweder wurde `\thesis` oder `\subject` mit einem speziellen Wert oder der Option `subjectthesis` verwendet.

Bezeichner	Deutsch	Englisch
<code>\datetext</code>	Eingereicht am	Submitted on
<code>\defensedatetext</code>	Verteidigt am	Defended on

`\abstractname` (Bezeichner)

Dieser Bezeichner wird lediglich für `tudscrbook` definiert, da dieser von KOMA-Script für die Buchklasse nicht vorgesehen wird.

Bezeichner	Deutsch	Englisch
<code>\abstractname</code>	Zusammenfassung	Abstract

v2.02

`\confirmationname` (Bezeichner)
`\blockingname` (Bezeichner)

Es werden die Bezeichnungen für Selbstständigkeitserklärung und Sperrvermerk für die dazugehörigen Überschriften definiert.

Bezeichner	Deutsch	Englisch
<code>\confirmationname</code>	Selbstständigkeitserklärung	Statement of authorship
<code>\blockingname</code>	Sperrvermerk	Restriction note

v2.02

`\confirmationtext` (Bezeichner)
`\blockingtext` (Bezeichner)

Die Texte der Erklärungen selbst sind derart aufgebaut, dass sie in Abhängigkeit von den angegebenen Informationen unterschiedlich ausgeführt werden. Innerhalb der Selbstständigkeitserklärung (`\confirmation`) werden gegebenenfalls die Felder für den Titel (`\title`) und die Typisierung der Abschlussarbeit⁴⁶ sowie die angegebenen Unterstützer⁴⁷ beachtet. Für den Sperrvermerk (`\blocking`) wird neben dem Titel (`\title`) optional außerdem noch das Feld der externen Firma⁴⁸ verwendet. Der Vollständigkeit halber werden im Folgenden noch die Texte für die Selbstständigkeitserklärung und den Sperrvermerk aufgeführt – allerdings lediglich die deutschsprachige Version. Dabei werden alle möglichen Felder angezeigt.

Bezeichner `\confirmationtext`

Hiermit versichere ich, dass ich die vorliegende *<Abschlussarbeit>* mit dem Titel *<Titel>* selbstständig und ohne unzulässige Hilfe Dritter verfasst habe. Es wurden keine anderen als die in der Arbeit angegebenen Hilfsmittel und Quellen benutzt. Die wörtlichen und sinngemäß übernommenen Zitate habe ich als solche kenntlich gemacht. Während der Anfertigung dieser Arbeit wurde ich nur von folgenden Personen unterstützt:

<Vorname Nachname>
<Vorname Nachname>

Weitere Personen waren an der geistigen Herstellung der vorliegenden Arbeit nicht beteiligt. Mir ist bekannt, dass die Nichteinhaltung dieser Erklärung zum nachträglichen Entzug des Hochschulabschlusses führen kann.

⁴⁶entweder `\thesis` oder `\subject{<Tabelle 2.1>}` beziehungsweise Option `subjectthesis=true`

⁴⁷`\confirmation[supporter=...]` oder `\supporter{...}`

⁴⁸`\blocking[company=...]` oder `\company{...}`

Bezeichner `\blockingtext`

Diese `<Abschlussarbeit>` mit dem Titel `<Titel>` enthält vertrauliche Informationen, offengelegt durch `<Firma>`. Veröffentlichungen, Vervielfältigungen und Einsichtnahme – auch nur auszugsweise – sind ohne ausdrückliche Genehmigung durch `<Firma>` nicht gestattet, ebenso wie Veröffentlichungen über den Inhalt dieser Arbeit. Die vorliegende Arbeit ist nur dem Betreuer an der Technischen Universität Dresden, den Gutachtern sowie den Mitgliedern des Prüfungsausschusses zugänglich zu machen.

<code>\coverpagename</code> (Bezeichner)
<code>\titlepagename</code> (Bezeichner)

Diese beiden Bezeichner werden bei aktivierter `tudbookmarks` für das Eintragen von Lesezeichen in ein PDF-Dokument genutzt.

Bezeichner	Deutsch	Englisch
<code>\coverpagename</code>	Umschlagseite	Cover page
<code>\titlepagename</code>	Titelblatt	Title page

<code>\listingname</code> (Bezeichner)
<code>\listlistingname</code> (Bezeichner)

Sollte ein Paket zur Einbindung von externem Quelltext – beispielsweise das Paket `listings` – verwendet werden, so werden diese Bezeichnungen für Quelltextausschnitte und das Quelltextverzeichnis verwendet.

Bezeichner	Deutsch	Englisch
<code>\listingname</code>	Quelltext	Listing
<code>\listlistingname</code>	Quelltextverzeichnis	List of Listings

2.5 KOMPATIBILITÄTSEINSTELLUNGEN ZU FRÜHEREN VERSIONEN

Bei der Entwicklung von TUD-KOMA-Script lässt es sich nicht immer vermeiden, dass Verbesserungen sowie Korrekturen an den Klassen und Paketen zu Änderungen am Ergebnis der Ausgabe führen, insbesondere bei Umbruch und Layout. Für bereits archivierte Dokumente, welche mit einer früheren Version erstellt wurden ist dies jedoch bei einer erneuten Kompilierung unter Umständen eher unerwünscht.

v2.03

<code>tudscrver=<Version> first last</code>	<code>preset: last</code>
---	---------------------------

Mit dieser Option wird es möglich, auf das (Umbruch-)Verhalten einer älteren respektive früheren Version von TUD-KOMA-Script umzuschalten, um nach der Kompilierung das erwartete Ergebnis zu erhalten. Neue Möglichkeiten, die sich nicht auf den Umbruch oder das Layout auswirken, sind auch für den Fall verfügbar, dass per Option die Kompatibilität zu einer älteren Version ausgewählt wurde.

Bei der Angabe einer unbekannten Version als Wert wird eine Warnung ausgegeben und `tudscrver=first` angenommen. Mit `tudscrver=last` wird die jeweils aktuell verfügbare Version ausgewählt und folglich auf die zukünftige Kompatibilität des Dokumentes zu der aktuell genutzten Version verzichtet. Dieses Verhalten entspricht der Voreinstellung. Es ist zu beachten, dass die Nutzung von `tudscrver` nur als Klassenoption möglich ist.

v2.03

2.02 || first:

Der Satzspiegel im Layout des Corporate Designs (*siehe cedgeometry*) wurde in der Version v2.03 leicht geändert. Der obere Seitenrand wurde verkleinert, der untere im gleichen Maße vergrößert. Der verfügbare Textbereich ist folglich identisch. Bei der Aktivierung des DRESDEN-concept-Logos im Fußbereich der Seite (*siehe ddcfoot*) wird im Gegensatz zur Version v2.02 der gleiche Satzspiegel genutzt. Mit `tudscrver=2.02` kann dieses Verhalten deaktiviert werden.

v2.04

2.03:

Seit der Version v2.04 werden mehrere Längen in Abhängigkeit der gewählten Schriftgröße definiert, welche durch die KOMA-Script-Option `fontsize` festgelegt wird. Zu beachten ist, dass diese *immer* als Klassenoption anzugeben ist. Von der gewählten Schriftgröße abhängig sind sowohl die dehnbaren Längen `\smallskipamount`, `\medskipamount` und `\bigskipamount` – welche von den Befehlen `\smallskip`, `\medskip` sowie `\bigskip` für das Einfügen vertikaler Abstände genutzt werden – als auch die beiden Längen `\abovecaptionskip` und `\belowcaptionskip` für den Abstand zwischen einem Gleitobjekt und dessen mit `\caption` gesetzten Beschreibung sowie `\columnsep` als Maß für den Abstand der einzelnen Textspalten im zwei- oder mehrspaltigen Layout. Mit der Wahl `tudscrver=2.03` lässt sich diese Funktionalität deaktivieren.

v2.05

2.04:

Mit der Version v2.05 werden die vorgegebenen Einstellungen zum Satzspiegel anhand der B-ISO/DIN-Reihe vorgenommen. Damit sind für alle Papierformate einer spezifischen ISO/DIN-Klasse die Seitenränder identisch. Mit der Wahl `tudscrver=2.04` ist der Satzspiegel von der A-ISO/DIN-Reihe abhängig, sodass die B- und C-Papierformate der gleichen Klasse größere Seitenränder erhalten, als die D- und A-Formate.

2.05:

Dies ist Kompatibilitätseinstellung für TUD-KOMA-Script v2.05 und wird für zukünftige Änderungen bereits vorgehalten. Soll ein mit der momentan aktuellen Version erzeugtes Dokument auch mit einer späteren Version von TUD-KOMA-Script nach einem \LaTeX -Lauf das gleiche Ausgabeergebnis liefern, muss dies mit `tudscrver=2.05` angegeben werden.

last:

Es werden keine Kompatibilitätseinstellungen für das Dokument vorgenommen. Mit einer späteren Version von TUD-KOMA-Script kann ein anderes Umbruchverhalten innerhalb des Dokumentes auftreten. Dies ist die Standardeinstellung.

KAPITEL 3

DIE POSTERKLASSE

Ergänzend zu den Hauptklassen, welche für das Setzen von Dokumenten im Corporate Design der Technischen Universität Dresden angeboten werden, wird die Klasse **tudscrposter** bereitgestellt. Mit dieser wird das Erstellen von Postern mit \LaTeX ermöglicht. Die Basis hierfür ist **tudscrartcl** und alle durch diese Klasse angebotenen Befehle und Optionen können mit **tudscrposter** gleichermaßen verwendet werden.

v2.05

`cdstyle=<Einstellung>` `preset: true`

`false/off/no:`

`true/on/yes/nocolor/monochrome:` (Säumniswert)

`lightcolor/pale:`

`barcolor:`

`bicolor/bichrome:`

`color:`

`full/fullcolor:`

v2.05

`\webpage{<Webseite>}`

v2.05

`\authorname` (Bezeichner)

`\contactname` (Bezeichner)

`\contactpersonname` (Bezeichner)

Bezeichner	Deutsch	Englisch
<code>\authorname</code>		
<code>\contactname</code>		
<code>\contactpersonname</code>	Ansprechpartner	Contact

KAPITEL 4

ZUSÄTZLICHE PAKETE IM TUD-KOMA-SCRIPT-BUNDLE

Zusätzlich zu den eigentlichen Hauptklassen werden im TUD-KOMA-Script-Bundle weitere Paket bereitgestellt. Diese werden im Folgenden vorgestellt.

4.1 DAS PAKET TUDSCRSUPERVISOR – STUDENTISCHE BETREUUNG

Dieses Paket stellt für das Erstellen von Aufgabenstellungen und Gutachten wissenschaftlicher Arbeiten sowie offiziellen Aushängen im Corporate Design passende Umgebungen und Befehle für den Anwender bereit. Deshalb richtet es sich vornehmlich an Mitarbeiter an der Technischen Universität Dresden, kann jedoch natürlich auch von Studenten genutzt werden.

4.1.1 AUFGABENSTELLUNG FÜR EINE WISSENSCHAFTLICHE ARBEIT

```
\begin{task}[<Überschrift> || <Parameterliste>] (tudpage in Abschnitt 2.2.2)
...
\end{task}
headline=<Überschrift> (Parameter)
```

Mit der `task`-Umgebung kann eine Aufgabenstellung für eine wissenschaftliche Arbeit ausgegeben werden. Diese basiert auf der Umgebung `tudpage` und akzeptiert deshalb im optionalen Argument alle Parameter, welche bei der Beschreibung von `tudpage` in [Abschnitt 2.2.2 auf Seite 29](#) erläutert wurden.

Für die Aufgabenstellung wird normalerweise eine Überschrift gesetzt, welche sich aus `\taskname` und – falls der Typ der Abschlussarbeit angegeben wurde – noch aus `\tasktext` und `\thesis` zusammensetzt. Der Parameter `headline` kann genutzt werden, um diese automatisch generierte Überschrift anzupassen.

Zu Beginn der Aufgabenstellung erscheint eine Tabelle mit den angegebenen Informationen zum Autor respektive zu den Autoren der Abschlussarbeit. Zwingend anzugeben sind dafür lediglich der Name des oder der Verfasser (`\author`) sowie der Titel der Arbeit (`\title`), welcher am Ende der Tabelle in fetter Schrift aufgeführt wird. Optional werden noch die Felder für den Studiengang (`\course`), die Fachrichtung (`\discipline`) sowie die Matrikelnummer (`\matriculationnumber`) und das Immatrikulationsjahr (`\matriculationyear`) ausgegeben, wobei nicht angegebene Felder bei der Ausgabe ignoriert werden. Der eigentliche Inhalt der Umgebung – sprich die Aufgabenstellung selbst – wird nach dem generierten Kopf ausgegeben.

Nach der Ausgabe des Inhaltes der Aufgabenstellung werden der oder die mit `\supervisor` definierten Betreuer aufgelistet. Dabei wird unter dem jeweiligen Namen selbst der sprachabhängige Bezeichner (`\supervisorname`, `\supervisorothername`) gesetzt. Darauf folgend erscheint das Ausgabedatum (`\issuedate`) und der verpflichtende Abgabetermin (`\duedate`). Zum Schluss wird die Unterschriftzeile für den Prüfungsausschussvorsitzenden (`\chairman`) und den betreuenden Hochschullehrer (`\professor`) gesetzt. Für genannte Personen werden unter dem Namen selbst die Bezeichner ausgegeben (`\chairmanname` und `\professorname`).

```
\taskform[<Parameterliste>]{<Ziele>}{<Schwerpunkte>}
```

Zusätzlich zur der frei gestaltbaren Umgebung `task` zur Erstellung einer Aufgabenstellung wird ein separater Befehl für eine standardisierte Ausgabe zur Verfügung gestellt. Dieser strukturiert die Aufgabenstellung in die zwei Bereiche *Ziele* und *Schwerpunkte* der Arbeit mit dazugehörigen Überschriften (`\objectivesname`, `\focusname`).

Im optionalen Argument können alle Parameter der Umgebung `task` verwendet werden. Im ersten obligatorischen Argument sollte ein Text mit einer kurzen thematischen Einordnung und dem eigentlichen Ziel der Arbeit erscheinen. Im zweiten Argument sollen die thematischen Schwerpunkte in Stichpunkten benannt werden. Der Inhalt des zweiten notwendigen

Argumentes wird in einer `itemize`-Umgebung gesetzt. Deshalb *muss* jedem Stichpunkt `\item` vorangestellt werden.

Beispiel: Die empfohlene Verwendung des Befehls `\taskform` ist wie folgt:

```
\taskform{%
  Motivation der Arbeit im ersten Absatz...

  Ziele der Arbeit im zweiten Absatz...
}{%
  \item Schwerpunkt 1
  \item Schwerpunkt 2
}
```

Hierzu sei auch auf das Minimalbeispiel in [Unterkapitel 5.4](#) verwiesen.

v2.02

```
\course{<Studiengang>}
\discipline{<Studienrichtung>}
```

Mit diesen beiden Befehlen kann der Studiengang sowie die Studienrichtung für den Autor oder die Autoren angegeben werden. Diese Informationen werden zu Beginn der `task`-Umgebung gesetzten Tabelle ausgegeben. Werden diese Befehle innerhalb des Makros `\author` verwendet, können auch unterschiedliche Angaben für mehrere Autoren gemacht werden. Dabei sind die Autoren mit `\and` voneinander zu trennen.

```
\chairman{<Prüfungsausschussvorsitzender>}
```

Wird dieses Feld genutzt, wird neben dem betreuenden Hochschullehrer (`\professor`) auch der Vorsitzende des Prüfungsausschusses am Ende der Aufgabenstellung aufgeführt. Dies wird zumeist für Abschlussarbeiten wie beispielsweise Master-Arbeit oder Diplomarbeit benötigt.

```
\issuedate{<Ausgabedatum>}
\duedate{<Abgabetermin>}
```

Mit diesen beiden Befehlen sollte das Datum der Ausgabe der Aufgabenstellung sowie der spätest mögliche Abgabetermin angegeben werden. Ist das Paket `isodate` geladen, wird die damit eingestellte Formatierung des Datums durch den Befehl `\printdate` aus diesem Paket für `\issuedate` und `\duedate` verwendet.

4.1.2 GUTACHTEN FÜR WISSENSCHAFTLICHE ARBEITEN

```
\begin{evaluation}[<Überschrift> || <Parameterliste>]
...
\end{evaluation}
headline=<Überschrift> (Parameter)
grade=<Note> (Parameter)
```

 (tudpage in [Abschnitt 2.2.2](#))

Diese Umgebung wird für das Erstellen eines Gutachtens einer wissenschaftlichen Arbeit bereitgestellt. Auch diese unterstützt alle Parameter, welche für die Umgebung `tudpage` in [Abschnitt 2.2.2 auf Seite 29](#) beschrieben wurden.

Für ein Gutachten wird gewöhnlich eine Überschrift aus `\evaluationname` und – falls der Abschlussarbeitstyp angegeben wurde – `\evaluationtext` sowie `\thesis` generiert. Diese

automatisch generierte Überschrift kann mit dem Parameter `headline` ersetzt werden. Am Ende des Gutachtens wird die mit `grade` gegebene Note in fester Schrift ausgezeichnet.

Am Anfang der `evaluation`-Umgebung wird die gleiche Tabelle mit Autorenangaben ausgegeben, wie dies bei der `task`-Umgebung der Fall ist. Nach dem Tabellenkopf folgt auch hier der eigentliche Inhalt, sprich das Gutachten der Abschlussarbeit. Abgeschlossen wird die Umgebung mit der gegebenen Note – welche innerhalb von `\gradetext` ausgegeben wird – sowie der Orts- und Datumsangabe (`\place`, `\date`) und der darauffolgenden Unterschriftzeile für den oder die Gutachter (`\referee`), welche wiederum mit den entsprechenden sprachabhängigen Bezeichner (`\refereename`, `\refereeothername`) ergänzt werden.

`\evaluationform[<Parameterliste>]{<Aufgabe>}{<Inhalt>}{<Bewertung>}{<Note>}`

Neben der individuell nutzbaren Umgebung `evaluation` wird ein separater Befehl zur Erstellung eines standardisierten Gutachtens bereitgestellt. Dieser strukturiert die Ausgabe in die vier Bereiche *Aufgabe*, *Inhalt*, *Bewertung* und *Note* und versieht diese jeweils mit der dazugehörigen Überschrift beziehungsweise Textausgabe (`\taskname`, `\contentname`, `\assessmentname` und `\gradetext`). Das optionale Argument unterstützt alle Parameter der `evaluation`-Umgebung.

Beispiel: Die empfohlene Verwendung des Befehls `\evaluationform` ist wie folgt:

```
\evaluationform{%
  Kurzbeschreibung der Aufgabenstellung...
}{%
  Zusammenfassung von Inhalt und Struktur...
}{%
  Bewertung der schriftlichen Abschlussarbeit...
}{%
  Zahl (Note)
}
```

Hierzu sei auch auf das Minimalbeispiel in [Unterkapitel 5.5](#) verwiesen.

`\grade{<Note>}`

Neben der Angabe der Note für ein Gutachten über den Parameter `grade` kann dafür auch dieser global wirkende Befehl verwendet werden.

4.1.3 AUSHANG

`\begin{notice}[<Überschrift> || <Parameterliste>]` (`tudpage` in [Abschnitt 2.2.2](#))
`...`
`\end{notice}`
`headline=<Überschrift> (Parameter)`

Für das Anfertigen eines Aushangs kann diese Umgebung verwendet werden. Diese basiert abermals auf der Umgebung `tudpage` und unterstützt alle deren Parameter.

Wurde ein Datum angegeben, wird dieses in der oberen rechten Ecke gesetzt. Anschließend wird die Überschrift ausgegeben, welche für gewöhnlich dem Inhalt von `\noticename` entspricht und mit `headline` geändert werden kann. Nach der Überschrift wird bereits der Inhalt der Umgebung ausgegeben. Wurde mit `\contactperson` ein oder mehrere Ansprechpartner angegeben, werden diese Informationen am Ende der Umgebung ausgegeben.

`\noticeform[<Parameterliste>]{<Inhalt>}{<Schwerpunkte>}`

Auch für diese Umgebung gibt es einen Befehl für eine normierte Form. Diese soll vor allem Verwendung für den Aushang studentischer Arbeitsthemen finden. Für das optionale Argument können sämtliche Parameter verwendet werden, die auch die `notice`-Umgebung unterstützt.

Das erste obligatorische Argument sollte für eine kurze Inhaltsbeschreibung verwendet werden. Neben dem textuellen Teil sollte hier wenn möglich eine thematisch passende Abbildung eingebunden werden (`\includegraphics`). Das zweite Argument wird –wie schon bei `\taskform`– dazu verwendet, einige Schwerpunkte aufzuzählen. Auch hier kommt nach der gliedernden Überschrift (`\focusname`) eine `itemize`-Umgebung zum Einsatz, allen Schwerpunkten muss ein `\item` vorangestellt werden.

Beispiel: Die empfohlene Verwendung des Befehls `\noticeform` ist wie folgt:

```
\noticeform{%
  Kurzbeschreibung des Inhaltes der studentischen Arbeit...

  Bild (optional), einzubinden mit:
  \includegraphics[<Optionen>]{<Datei>}
}%
  \item Schwerpunkt 1
  \item Schwerpunkt 2
}
```

Hierzu sei auch auf das Minimalbeispiel in [Unterkapitel 5.6](#) verwiesen.

v2.02

```
\contactperson{<Kontaktperson(en)>}
\office{<Dienststz>}
\telephone{<Telefonnummer>}
\emailaddress{<E-Mail-Adresse>}
```

Am Ende eines Aushangs können mit `\contactperson` Kontaktinformationen für eine oder mehrere Ansprechpartner angegeben werden. Soll mehr als eine Kontaktperson genannt werden, so müssen diese innerhalb des Befehls `\contactperson` mit dem Befehl `\and` getrennt werden. Für jede Person kann innerhalb des Argumentes von `\contactperson` der Dienststz (`\office`), die dienstliche Telefonnummer (`\telephone`) sowie die geschäftliche E-Mail-Adresse (`\emailaddress`) angegeben werden. Sollte das Paket `hyperref` geladen werden, wird die gegebene E-Mail-Adresse direkt in einen entsprechenden Link gewandelt.

4.1.4 ZUSÄTZLICHE SPRACHABHÄNGIGE BEZEICHNER

Für das Paket `tudscrsupervisor` werden für die zusätzlichen Befehle und Umgebungen weitere Bezeichner definiert. Für eine etwaige Anpassung dieser sei auf [Unterkapitel 2.4](#) verwiesen.

```
\taskname (Bezeichner)
\tasktext (Bezeichner)
```

Die Bezeichnung der Aufgabenstellung selbst ist in `\taskname` enthalten. Für die Generierung einer Überschrift wird dieser verwendet. Wurde außerdem mit `\thesis` oder `\subject` der Typ der Abschlussarbeit⁴⁹ angegeben, wird die Überschrift zusammen mit dem Bezeichner

⁴⁹`subjectthesis` oder spezieller Wert aus [Tabelle 2.1](#)

`\tasktext` um die Typisierung erweitert. Falls gewünscht, kann die automatisch generierte Überschrift mit dem Parameter `headline` der Umgebung `task` überschrieben werden.

Bezeichner	Deutsch	Englisch
<code>\taskname</code>	Aufgabenstellung	Task
<code>\tasktext</code>	für die Anfertigung einer	for the preparation of a

v2.02
v2.04

`\namesname` (Bezeichner)
`\titlename` (Bezeichner)
`\coursename` (Bezeichner)
`\disciplinename` (Bezeichner)

Diese Bezeichner werden in der Tabelle mit den Autoreninformationen zu Beginn der Aufgabenstellung verwendet. Dabei werden `\coursename` und `\disciplinename` nur genutzt, wenn für mindestens einen Autor die Befehle `\course` und/oder `\discipline` verwendet wurden.

Bezeichner	Deutsch	Englisch
<code>\namesname</code>	Name	Name
<code>\titlename</code>	Titel	Title
<code>\coursename</code>	Studiengang	Course
<code>\disciplinename</code>	Studienrichtung	Discipline

`\issuedatetext` (Bezeichner)
`\duedatetext` (Bezeichner)

Am Ende der Aufgabenstellung wird nach dem oder der Betreuer das Ausgabe- und Abgabedatum (`\issuedate`, `\duedate`) der Abschlussarbeit mit folgenden Bezeichner erläutert.

Bezeichner	Deutsch	Englisch
<code>\issuedatetext</code>	Ausgehändigt am	Issued on
<code>\duedatetext</code>	Einzureichen am	Due date for submission

`\chairmanname` (Bezeichner)

Wurde der Prüfungsausschussvorsitzende (`\chairman`) angegeben, erfolgt unter dem Namen selbst die Ausgabe des Bezeichners.

Bezeichner	Deutsch	Englisch
<code>\chairmanname</code>	Prüfungsausschussvorsitzender	Chairman of the Audit Committee

`\focusname` (Bezeichner)
`\objectivesname` (Bezeichner)

Die Vorlagen für Aufgabenstellung (`\taskform`) beziehungsweise Aushang (`\noticeform`) nutzen für die gesetzten Überschriften diese Bezeichner.

Bezeichner	Deutsch	Englisch
<code>\focusname</code>	Schwerpunkte der Arbeit	Focus of work
<code>\objectivesname</code>	Ziele der Arbeit	Objectives of work

\evaluationname (Bezeichner)
\evaluationtext (Bezeichner)

Die Bezeichnung des Gutachten selbst ist in \evaluationname enthalten. Für die Generierung der Überschrift wird der Bezeichner \evaluationtext sowie der mit \thesis oder gegebenenfalls mit \subject gegebenen Typ der Abschlussarbeit verwendet. Diese automatisch generierte Überschrift kann mit dem Parameter headline der Umgebung evaluation durch den Anwender überschrieben werden.

Bezeichner	Deutsch	Englisch
\evaluationname	Gutachten	Evaluation
\evaluationtext	für die	for the

\contentname (Bezeichner)
\assessmentname (Bezeichner)

Bei der standardisierten Form des Gutachten (\evaluationform) werden die darin – zur strukturierter Gliederung – erzeugten Überschriften mit den Bezeichnern \taskname, \contentname und \assessmentname gesetzt.

Bezeichner	Deutsch	Englisch
\taskname	Aufgabenstellung	Task
\contentname	Inhalt	Content
\assessmentname	Bewertung	Assessment

\gradetext (Bezeichner)

Wird für das Gutachten einer wissenschaftlichen Arbeit die erzielte Note entweder mit dem Befehl \grade{<Note>} oder alternativ dazu mit dem Parameter grade=<Note> angegeben, so wird diese innerhalb von \gradetext verwendet.

Bezeichner	Deutsch	Englisch
\gradetext	Die Arbeit wird mit der Note <Note> bewertet.	The thesis is evaluated with a grade of <Note>.

v2.02

\noticename (Bezeichner)
\contactpersonname (Bezeichner)

Die Bezeichnung des Aushangs selbst ist in \noticename enthalten. Für die Generierung einer Überschrift wird dieser verwendet. Falls gewünscht, kann diese mit dem Parameter headline der Umgebung notice überschrieben werden. Wurde eine Kontaktperson mit dem Befehl \contactperson angegeben, wird als Überschrift der Kontaktdaten der Bezeichner \contactpersonname verwendet.

Bezeichner	Deutsch	Englisch
\noticename	Aushang	Notice
\contactpersonname	Ansprechpartner	Contact

4.2 DAS PAKET TUDSCRCOLOR – FARBEN IM CORPORATE DESIGN

Zur Verwendung der Farben des Corporate Designs wird das Paket `tudscrcolor` genutzt. Falls dieses nicht in der Präambel geladen wird – um beispielsweise zusätzliche Optionen aufzurufen – binden die TUD-KOMA-Script-Klassen dieses automatisch ein.

Für das Corporate Design sind mehrere Farben vorgesehen. Die prägnanteste aller ist die Hausfarbe `HKS41`, danach folgen die Farben für Auszeichnungen der ersten (`HKS44`) und der zweiten Kategorie (`HKS36`, `HKS33`, `HKS57`, `HKS65`) sowie eine Ausnahmefarbe (`HKS07`). Diese Farben dürfen sowohl in ihrer Grundform als auch in helleren Tönen mit einer Abstufung in 10 %-Schritten verwendet werden. Das ohnehin verwendete Paket `xcolor` stellt genau diese Funktionalität zur Verfügung. Jede der Farben kann sowohl über `HKS<Zahl>` als auch über ein Pseudonym `cd<Farbe>` genutzt werden.

Beispiel: Die Grundfarbe `HKS44` soll in der auf 20% reduzierten, helleren Abstufung genutzt werden. Innerhalb eines Befehls, der als Argument eine gültige Farbe erwartet, muss lediglich `HKS44!20` angegeben werden. Dies wird hier exemplarisch mit dieser `Box \colorbox{HKS44!20}{Box}` demonstriert.

Bei der farbigen Gestaltung des Corporate Designs (`cd=color`) ist der Hintergrund von Umschlagseite, Titel sowie Teilen in `HKS41` und die Schrift auf selbigen in `HKS41!30` gehalten. Der Kapitelseitenhintergrund erscheint in `HKS41!10`, die Schrift in `HKS41`. Bei geringerem Farbeinsatz werden lediglich die Schriften der Gliederungsseiten auf `HKS41` gesetzt.

Sollen bestimmte Optionen an das Paket `xcolor` weitergereicht werden, gibt es dafür zwei Möglichkeiten. Diese kann entweder vor dem Laden der Klasse direkt an `xcolor` übergeben werden⁵⁰ oder es wird `tudscrcolor` mit der entsprechenden Option geladen.⁵¹

4.2.1 GENERELLE FARBDEFINITIONEN

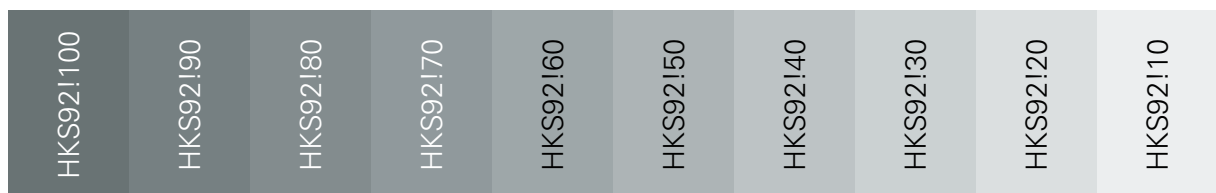
PRIMÄRE HAUSFARBE

HKS41 (cddarkblue) (Farbe)



SEKUNDÄRE HAUSFARBE (GESCHÄFTSAUSSTATTUNG)

HKS92 (cdgray) (Farbe)



⁵⁰ `\PassOptionsToPackage{<Option>}{xcolor}` gefolgt von `\documentclass[<Klassenoptionen>]{tudscr...}`

⁵¹ `\usepackage[<Option>]{tudscrcolor}`; `tudscrcolor` reicht `<Option>` an `xcolor` weiter

AUSZEICHNUNGSFARBE 1.KATEGORIE

HKS44 (cdblue) (Farbe)

HKS44i100	HKS44i90	HKS44i80	HKS44i70	HKS44i60	HKS44i50	HKS44i40	HKS44i30	HKS44i20	HKS44i10
-----------	----------	----------	----------	----------	----------	----------	----------	----------	----------

AUSZEICHNUNGSFARBE 2.KATEGORIE

HKS36 (cdindigo) (Farbe)
HKS33 (cdpurple) (Farbe)
HKS57 (cddarkgreen) (Farbe)
HKS65 (cdgreen) (Farbe)

HKS36i100	HKS36i90	HKS36i80	HKS36i70	HKS36i60	HKS36i50	HKS36i40	HKS36i30	HKS36i20	HKS36i10
-----------	----------	----------	----------	----------	----------	----------	----------	----------	----------

HKS33i100	HKS33i90	HKS33i80	HKS33i70	HKS33i60	HKS33i50	HKS33i40	HKS33i30	HKS33i20	HKS33i10
-----------	----------	----------	----------	----------	----------	----------	----------	----------	----------

HKS57i100	HKS57i90	HKS57i80	HKS57i70	HKS57i60	HKS57i50	HKS57i40	HKS57i30	HKS57i20	HKS57i10
-----------	----------	----------	----------	----------	----------	----------	----------	----------	----------

HKS65i100	HKS65i90	HKS65i80	HKS65i70	HKS65i60	HKS65i50	HKS65i40	HKS65i30	HKS65i20	HKS65i10
-----------	----------	----------	----------	----------	----------	----------	----------	----------	----------

AUSNAHMEFARBE

HKS07 (cdorange) (Farbe)

HKS07i100	HKS07i90	HKS07i80	HKS07i70	HKS07i60	HKS07i50	HKS07i40	HKS07i30	HKS07i20	HKS07i10
-----------	----------	----------	----------	----------	----------	----------	----------	----------	----------

4.2.2 ZUSÄTZLICHE FARBDEFINITIONEN

Das Paket `tudscrcolor` definiert im Normalfall lediglich die zuvor beschriebenen Grundfarben `HKS41`, `HKS92`, `HKS44`, `HKS36`, `HKS33`, `HKS57`, `HKS65` sowie `HKS07`. Alle anderen farblichen Abstufungen können mit den beschriebenen Möglichkeiten des Paketes `xcolor` generiert werden.

oldcolors

In den letzten Jahren sind viele verschiedene Klassen und Pakete für das Corporate Design der Technischen Universität Dresden entstanden. Innerhalb dieser existieren abweichende Farbdefinitionen. Um eine Migration von den benannten Klassen und Paketen auf TUD-KOMA-Script zu ermöglichen, existiert die Paketoption `oldcolors`. Wird diese genutzt, so werden zusätzliche Farben nach dem Schema `HKS41K<Zahl>` und `HKS41-<Zahl>` definiert, wobei der hinten angestellte Zahlenwert aus der 10er-Reihe kommen muss.

4.2.3 UMSTELLUNG DES FARBMODELLS

Normalerweise verwendet `tudscrcolor` das CMYK-Farbmodell. Außerdem wird weiterhin noch der RGB-Farbraum unterstützt. Eine Umschaltung des Farbmodells ist beispielsweise für gewisse Funktionen des Paketes `tikz` notwendig.

RGB

Mit dieser Option werden bereits beim Laden des Paketes `tudscrcolor` die Farben nicht nach dem CMYK-Farbmodell sondern im RGB-Farbraum global definiert.

`\setcdcolors{<Farbmodell>}`

Mit diesem Befehl kann innerhalb des Dokumentes das verwendete Farbmodell angepasst werden. Damit ist es möglich, lokal innerhalb einer Umgebung den Farbmodus zu ändern und so nur in bestimmten Situationen beispielsweise aus dem CMYK-Farbmodell in den RGB-Farbraum zu wechseln. Unterstützte Werte für `<Farbmodell>` sind CMYK und RGB beziehungsweise `rgb`.

Achtung!

Beachten Sie, dass die Darstellung der Farben im jeweiligen Farbmodus (CMYK oder RGB) je nach verwendeter Bildschirm- Drucker- und Softwarekonfiguration verschieden ausfallen kann. Die verwendeten RGB-Werte entstammen aus dem Handbuch zum Corporate Design und sind lediglich Näherungswerte. Abweichungen vom gedruckten HKS-Farbregister und selbst ermittelten Werten sind technisch nicht zu vermeiden.

4.3 DAS PAKET TUDSCRFONTS – SCHRIFTEN IM CORPORATE DESIGN

v2.02

Dieses Paket stellt die Schriften des Corporate Designs für $\text{\LaTeX} 2_{\epsilon}$ -Klassen bereit, welche *nicht* zum TUD-KOMA-Script-Bundle gehören. Das Paket unterstützt einen Großteil der in [Unterkapitel 2.1 auf Seite 16](#) beschriebenen Optionen und Befehle. Die nutzbaren Paketoptionen sind für den Fließtext `cdfont` – ohne die Einstellungsmöglichkeiten für den Querbalken des Corporate Designs (`cdhead`) – und für die mathematischen Schriften `cdmath` sowie `slantedgreek`. Da von `tudscrfonts` intern das Paket `tudscrbase` geladen wird, können diese entweder als Paketoptionen im optionalen Argument von `\usepackage[<Paketoption>]{tudscrfonts}` oder direkt als Klassenoption angegeben werden. Zusätzlich ist nach dem Laden des Paketes die spätere Optionenwahl mit `\TUDoption` beziehungsweise `\TUDOptions` möglich.

Des Weiteren wird das Paket `textcase` geladen, welches die Befehle `\MakeTextUppercase` und `\NoCaseChange` zur Verfügung stellt. Der Befehl `\ifdin` wird ebenso wie die in [Unterkapitel 2.1](#) beschriebenen Textschalter und -kommandos sowie die Befehle für die griechischen Buchstaben bereitgestellt.

v2.04

Das Paket `tudscrfonts` ist insbesondere für die Verwendung zusammen mit einer der Klassen `tudbook`, `tudbeamer`, `tudletter`, `tudfax`, `tudhaus` sowie `tudform` vorgesehen. Zusätzlich werden seit der Version v2.04 auch `tudmathposter` und `tudposter` unterstützt. Das Paket kann prinzipiell mit jeder beliebigen \LaTeX 2_ε-Klasse verwendet werden. Wird keine der zuvor genannten Klassen genutzt, muss der Anwender das Setzen der Gliederungsüberschriften in Majuskeln der **DIN BOLD** – wie es im Corporate Design vorgesehen ist – selbst umsetzen. Dafür sei abermals auf die Textauswahlbefehle `\textdbn` und `\dinbn` sowie den Befehl `\MakeTextUppercase` zur automatisierten Großschreibung hingewiesen.

Die Schriftinstallation für das TUD-KOMA-Script-Bundle unterscheidet sich von der für die gerade genannten Klassen sehr stark. Dabei wurde auch die Bezeichnung der Schriftfamilien geändert. Dies hatte zwei Gründe, wobei der letztere der entscheidende ist:

1. Die bisherige Schriftbenennung entsprach nicht dem offiziellen [T_EX-Namensschema](#)⁵²
2. Bei der Installation für das TUD-KOMA-Script-Bundle werden sowohl die Metriken als auch das Kerning der Schriften für Fließtext und den Mathematikmodus angepasst, was das Ergebnis der erzeugten Ausgabe beeinflusst. Damit jedoch Dokumente, die mit den zuvor genannten, älteren Klassen erstellt wurden, weiterhin genauso ausgegeben werden wie bisher, mussten die Schriftfamilien einen neuen Namen erhalten.

Wird nun das Paket `tudscrfonts` einer der zuvor genannten, älteren Klassen verwendet, hat dies den Vorteil, dass auch in diesen das angepasste Kerning der Schriften sowie der stark verbesserte Mathematiksatz zum Tragen kommen. Außerdem kann bei der Verwendung von `tudscrfonts` auf eine Installation der Schriften des Corporate Designs in der alten Variante verzichtet werden. *In diesem Fall kann sich das Ausgabeergebnis im Vergleich zu der Varianten mit den alten Schriften ändern. Alternativ zur Verwendung des Paketes `tudscrfonts` können die alten Schriftfamilien auch parallel zu den neuen installiert werden. Hierfür werden die Skripte `tudfonts_install.bat`⁵³ beziehungsweise `tudfonts_install.sh`⁵⁴ bereitgestellt.*

Achtung!

4.4 DAS PAKET MATHSWAP

Die Verwendung von Dezimal- und Tausendertrennzeichen im mathematischen Satz sind regional sehr unterschiedlich. In den meisten englischsprachigen Ländern wird der Punkt als Dezimaltrennzeichen und das Komma zur Zifferngruppierung verwendet, im restlichen Europa wird dies genau entgegengesetzt praktiziert. Dieses Paket soll dazu dienen, beliebige formatierte Zahlen in ihrer Ausgabe anzupassen. Dafür werden die Zeichen Punkt (.) und Komma (,) als aktive Zeichen im Mathematikmodus definiert.

Ähnliche Funktionalitäten werden bereits durch die Pakete `icomma` und `ziffer` bereitgestellt. Bei `icomma` muss jedoch beim Verfassen des Dokumentes durch den Autor beachtet werden, ob das verwendete Komma einem Dezimaltrennzeichen entspricht ($t = 1,2$) oder einem normalen Komma im Mathematiksatz ($z = f(x, y)$), wo ein gewisser Abstand nach dem Komma

⁵²<http://mirrors.ctan.org/info/fontname/fontname.pdf>

⁵³https://github.com/tud-cd/tudscrold/releases/download/fonts/tudfonts_install.bat

⁵⁴https://github.com/tud-cd/tudscrold/releases/download/fonts/tudfonts_install.sh

durchaus gewünscht ist. Das Paket **ziffer** liefert dafür die gewünschte Funktionalität,⁵⁵ ist allerdings etwas unflexibel, was den Umgang mit den Trennzeichen anbelangt. Als Alternative zu diesem Paket kann außerdem **ionumbers** verwendet werden.

Das Paket **mathswap** sorgt dafür, dass Trennzeichen direkt vor einer Ziffer erkannt und nach bestimmten Vorgaben ersetzt werden. Sollte sich jedoch zwischen Trennzeichen und Ziffer Leerraum befinden, wird dieser als solcher auch gesetzt. Für ein Beispiel zur Verwendung des Paketes sei auf das Tutorial **mathswap.pdf** in **Unterkapitel 5.9** hingewiesen.

```
\commaswap{<Trennzeichen>}
\dotswap{<Trennzeichen>}
```

Die beiden Befehle `\commaswap` und `\dotswap` sind die zentrale Benutzerschnittstelle des Paketes. Das Makro `\commaswap` definiert das Trennzeichen oder den Inhalt, wodurch ein Komma ersetzt werden soll, auf welches direkt danach eine Ziffer folgt. Normalerweise setzt \LaTeX nach einem Komma im mathematischen Satz zusätzlich einen horizontalen Abstand. Bei der Ersetzung durch `\commaswap` entfällt dieser. Die Voreinstellung für `\commaswap` ist deshalb auf ein Komma (,) gesetzt. Mit dem Makro `\dotswap` kann definiert werden, wodurch der Punkt im mathematischen Satz ersetzt werden soll, wenn auf diesen direkt anschließend eine Ziffer folgt. Da der Punkt im deutschsprachigem Raum zur Gruppierung von Ziffern genutzt wird, ist hierfür standardmäßig ein halbes geschütztes Leerzeichen definiert (`\,`).

v2.02

```
\mathswapon
\mathswapoff
```

Die Funktionalität von **mathswap** kann innerhalb des Dokumentes mit diesen beiden Befehlen an- und abgeschaltet werden. Beim Laden des Paketes ist es standardmäßig aktiviert.

4.5 DAS PAKET TWOCOLFIX

Der $\text{\LaTeX} 2_{\epsilon}$ -Kernel enthält einen Fehler, der Kapitelüberschriften im zweispaltigen Layout höher setzt, als im einspaltigen. Der **Fehler**⁵⁶ ist zwar schon länger bekannt, allerdings bisher noch nicht in den $\text{\LaTeX} 2_{\epsilon}$ -Kernel übernommen worden. Das Paket **twocolfix** behebt das Problem. Eine Integration dieses Bugfixes in KOMA-Script wurde bereits bei Markus Kohm angefragt, jedoch von ihm bis jetzt **nicht weiter verfolgt**⁵⁷.

ZUKÜNFTIGE ARBEITEN

Diese Dinge sollen langfristig in das TUD-KOMA-Script-Bundle eingearbeitet werden:

Es soll das Paket **tudscrletter** für Briefe im Corporate Design der Technischen Universität Dresden entstehen. Auch Vorlagen für Fax und Hausmitteilungen sollen dabei abfallen. Mit **tudscrbeamer** soll ein Paket entwickelt werden, mit dem sich \LaTeX -Beamer-Präsentationen im Stil der Technischen Universität Dresden erstellt werden können. Außerdem ist ein Paket **tudscrlayout** vorstellbar, mit dem entweder die **tudheadings**-Seitenstile in anderen Klassen genutzt werden können oder der durch das Corporate Design vorgegebene Satzspiegel ohne den Seitenstil selbst verwendet wird, um vorgedrucktes Papier verwenden zu können.

⁵⁵kein Leerraum nach Komma, wenn direkt danach eine Ziffer folgt

⁵⁶<http://www.latex-project.org/cgi-bin/ltxbugs2html?pr=latex/3126>

⁵⁷<http://www.komascript.de/node/1681>



TEIL II

ERGÄNZUNGEN UND HINWEISE

Für die Verwendung des TUD-KOMA-Script-Bundles ist es nicht notwendig, diesen Teil zu lesen. In **Kapitel 5** sind insbesondere für \LaTeX -Neulinge sowie neue Anwender des TUD-KOMA-Script-Bundles mehrere einfache Beispiele sowie umfangreichere Tutorials für dessen Verwendung zu sehen. In **Kapitel 6** werden Einsteigern – und auch dem bereits versierten \LaTeX -Nutzer – meiner Meinung nach empfehlenswerte Pakete kurz vorgestellt.

Anwendungshinweise sowie der eine oder andere allgemeine Hinweis bei der Verwendung von $\LaTeX 2_{\epsilon}$ wird in **Kapitel 7** gegeben. Dabei sind diese durchaus für die Verwendung sowohl des TUD-KOMA-Script-Bundles als auch anderer \LaTeX -Klassen interessant. Für Anregungen, Hinweise, Ratschläge oder Empfehlungen zu weiteren Paketen sowie Tipps bin ich jederzeit empfänglich.

KAPITEL 5

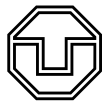
MINIMALBEISPIELE UND TUTORIALS

Dieses Kapitel soll den Einstieg und den ersten Umgang mit TUD-KOMA-Script erleichtern. Dafür werden einige Minimalbeispiele gegeben, die einzelne Funktionalitäten darstellen. Diese sind so reduziert ausgeführt, dass sie sich dem Anwender direkt erschließen sollten. Des Weiteren werden weiterführende und kommentierte Anwendungsbeispiele bereitgestellt. Diese Tutorials sind nicht unmittelbar im Handbuch enthalten sondern werden als externe Dateien bereitgehalten, welche direkt via Hyperlink geöffnet werden können.

5.1 DOKUMENT

Hier wird gezeigt, wie die Präambel eines minimalen $\text{\LaTeX} 2_{\epsilon}$ -Dokumentes gestaltet werden sollte. Dieser Ausschnitt kann prinzipiell als Grundlage für ein neu zu erstellendes Dokument verwendet werden. Lediglich das Einbinden des Paketes **blindtext** mit `\usepackage{blindtext}` und die Verwendung des daraus stammenden Befehls `\blinddocument` können weggelassen werden.

```
\RequirePackage[ngerman=ngerman-x-latest]{hyphsubst}
\documentclass[ngerman]{tudscrreprt}
\usepackage{selinput}\SelectInputMappings{adieresis={ä},germandbls={ß}}
\usepackage[T1]{fontenc}
\usepackage{babel}
\usepackage{blindtext}
\begin{document}
\faculty{Juristische Fakultät}
\department{Fachrichtung Strafrecht}
\institute{Institut für Kriminologie}
\chair{Lehrstuhl für Kriminalprognose}
\date{18.02.2015}
\author{Mickey Mouse}
\title{Der Geldspeicher in Entenhausen}
\maketitle
\blinddocument
\end{document}
```



DER GELDSPEICHER IN ENTENHAUSEN

Mickey Mouse

18.02.2015

1 ÜBERSCHRIFT AUF EBENE 0 (CHAPTER)

Dies hier ist ein Blindtext zum Testen von Textausgaben. Wer diesen Text liest, ist selbst schuld. Der Text gibt lediglich den Grauwert der Schrift an. Ist das wirklich so? Ist es gleichgültig, ob ich schreibe: „Dies ist ein Blindtext“ oder „Huardest gefburn“? Kjift – mitnichten! Ein Blindtext bietet mir wichtige Informationen. An ihm messe ich die Lesbarkeit einer Schrift, ihre Anmutung, wie harmonisch die Figuren zueinander stehen und prüfe, wie breit oder schmal sie läuft. Ein Blindtext sollte möglichst viele verschiedene Buchstaben enthalten und in der Originalsprache gesetzt sein. Er muss keinen Sinn ergeben, sollte aber lesbar sein. Fremdsprachige Texte wie „Lorem ipsum“ dienen nicht dem eigentlichen Zweck, da sie eine falsche Anmutung vermitteln.

1.1 ÜBERSCHRIFT AUF EBENE 1 (SECTION)

Dies hier ist ein Blindtext zum Testen von Textausgaben. Wer diesen Text liest, ist selbst schuld. Der Text gibt lediglich den Grauwert der Schrift an. Ist das wirklich so? Ist es gleichgültig, ob ich schreibe: „Dies ist ein Blindtext“ oder „Huardest gefburn“? Kjift – mitnichten! Ein Blindtext bietet mir wichtige Informationen. An ihm messe ich die Lesbarkeit einer Schrift, ihre Anmutung, wie harmonisch die Figuren zueinander stehen und prüfe, wie breit oder schmal sie läuft. Ein Blindtext sollte möglichst viele verschiedene Buchstaben enthalten und in der Originalsprache gesetzt sein. Er muss keinen Sinn ergeben, sollte aber lesbar sein. Fremdsprachige Texte wie „Lorem ipsum“ dienen nicht dem eigentlichen Zweck, da sie eine falsche Anmutung vermitteln.

1.1.1 ÜBERSCHRIFT AUF EBENE 2 (SUBSECTION)

Dies hier ist ein Blindtext zum Testen von Textausgaben. Wer diesen Text liest, ist selbst schuld. Der Text gibt lediglich den Grauwert der Schrift an. Ist das wirklich so? Ist es gleichgültig, ob ich schreibe: „Dies ist ein Blindtext“ oder „Huardest gefburn“? Kjift – mitnichten! Ein Blindtext bietet mir wichtige Informationen. An ihm messe ich die Lesbarkeit einer Schrift, ihre Anmutung, wie harmonisch die Figuren zueinander stehen und prüfe, wie breit oder schmal sie läuft. Ein Blindtext sollte möglichst viele verschiedene Buchstaben enthalten und in der Originalsprache gesetzt sein. Er muss keinen Sinn ergeben, sollte aber lesbar sein. Fremdsprachige Texte wie „Lorem ipsum“ dienen nicht dem eigentlichen Zweck, da sie eine falsche Anmutung vermitteln.

ÜBERSCHRIFT AUF EBENE 3 (SUBSUBSECTION)

Dies hier ist ein Blindtext zum Testen von Textausgaben. Wer diesen Text liest, ist selbst schuld. Der Text gibt lediglich den Grauwert der Schrift an. Ist das wirklich so? Ist es gleichgültig, ob ich schreibe: „Dies ist ein Blindtext“ oder „Huardest gefburn“? Kjift – mitnichten! Ein Blindtext bietet mir wichtige Informationen. An ihm messe ich die Lesbarkeit einer Schrift, ihre Anmutung, wie harmonisch die Figuren zueinander stehen und prüfe, wie breit oder schmal sie läuft. Ein Blindtext sollte möglichst viele verschiedene Buchstaben enthalten und in der Originalsprache gesetzt sein. Er muss keinen Sinn ergeben, sollte aber lesbar sein. Fremdsprachige Texte wie „Lorem ipsum“ dienen nicht dem eigentlichen Zweck, da sie eine falsche Anmutung vermitteln.

Überschrift auf Ebene 4 (paragraph) Dies hier ist ein Blindtext zum Testen von Textausgaben. Wer diesen Text liest, ist selbst schuld. Der Text gibt lediglich den Grauwert der Schrift an. Ist das wirklich so? Ist es gleichgültig, ob ich schreibe: „Dies ist ein Blindtext“ oder „Huardest gefburn“? Kjift – mitnichten! Ein Blindtext bietet mir wichtige Informationen. An ihm messe ich die Lesbarkeit einer Schrift, ihre Anmutung, wie harmonisch die Figuren zueinander stehen und prüfe, wie breit oder schmal sie läuft. Ein Blindtext sollte möglichst viele verschiedene Buchstaben enthalten und in der Originalsprache gesetzt sein. Er muss keinen Sinn ergeben, sollte aber lesbar sein. Fremdsprachige Texte wie „Lorem ipsum“ dienen nicht dem eigentlichen Zweck, da sie eine falsche Anmutung vermitteln.

1.2 LISTEN

1.2.1 BEISPIEL EINER LISTE (ITEMIZE)

- Erster Listenpunkt, Stufe 1
- Zweiter Listenpunkt, Stufe 1
- Dritter Listenpunkt, Stufe 1
- Vierter Listenpunkt, Stufe 1
- Fünfter Listenpunkt, Stufe 1

BEISPIEL EINER LISTE (4*ITEMIZE)

- Erster Listenpunkt, Stufe 1
 - Erster Listenpunkt, Stufe 2
 - * Erster Listenpunkt, Stufe 3
 - Erster Listenpunkt, Stufe 4
 - Zweiter Listenpunkt, Stufe 4
 - * Zweiter Listenpunkt, Stufe 3
 - Zweiter Listenpunkt, Stufe 2
- Zweiter Listenpunkt, Stufe 1

1.2.2 BEISPIEL EINER LISTE (ENUMERATE)

1. Erster Listenpunkt, Stufe 1
2. Zweiter Listenpunkt, Stufe 1
3. Dritter Listenpunkt, Stufe 1
4. Vierter Listenpunkt, Stufe 1
5. Fünfter Listenpunkt, Stufe 1

BEISPIEL EINER LISTE (4*ENUMERATE)

1. Erster Listenpunkt, Stufe 1
 - a) Erster Listenpunkt, Stufe 2
 - i. Erster Listenpunkt, Stufe 3
 - A. Erster Listenpunkt, Stufe 4
 - B. Zweiter Listenpunkt, Stufe 4
 - ii. Zweiter Listenpunkt, Stufe 3
 - b) Zweiter Listenpunkt, Stufe 2
2. Zweiter Listenpunkt, Stufe 1

1.2.3 BEISPIEL EINER LISTE (DESCRIPTION)

Erster Listenpunkt, Stufe 1

Zweiter Listenpunkt, Stufe 1

Dritter Listenpunkt, Stufe 1

Vierter Listenpunkt, Stufe 1

Fünfter Listenpunkt, Stufe 1

BEISPIEL EINER LISTE (4*DESCRIPTION)

Erster Listenpunkt, Stufe 1

Erster Listenpunkt, Stufe 2

Erster Listenpunkt, Stufe 3

Erster Listenpunkt, Stufe 4

Zweiter Listenpunkt, Stufe 4

Zweiter Listenpunkt, Stufe 3

Zweiter Listenpunkt, Stufe 2

Zweiter Listenpunkt, Stufe 1

5.2 DISSERTATION

Eine Abschlussarbeit oder ähnliches könnte wie hier gezeigt begonnen werden.

```
\RequirePackage[ngerman=ngerman-x-latest]{hyphsubst}
\documentclass[english,ngerman,BCOR=6mm,cdgeometry=no,DIV=13]{tudscrreprt}
\usepackage{selinput}\SelectInputMappings{adieresis={ä},germandbls={ß}}
\usepackage[T1]{fontenc}
\usepackage{babel}
\usepackage{isodate}
\usepackage{blindtext}
\begin{document}
\faculty{Juristische Fakultät}
\department{Fachrichtung Strafrecht}
\institute{Institut für Kriminologie}
\chair{Lehrstuhl für Kriminalprognose}
\date{18.02.2015}
\author{Mickey Mouse}
\title{%
  Entwicklung eines optimalen Verfahrens zur Eroberung des
  Geldspeichers in Entenhausen
}
\thesis{diss}
\graduation[Dr. jur.]{Doktor der Rechte}
\dateofbirth{2.1.1990}
\placeofbirth{Dresden}
\defensedate{15.03.2015}
\referee{Dagobert Duck \and Mac Moneysac}
\maketitle
\TUDoption{abstract}{section,multiple}
\begin{abstract}[pagestyle=empty.tudheadings]
\blindtext
\nextabstract[english]
\blindtext
\end{abstract}
\end{document}
```



**TECHNISCHE
UNIVERSITÄT
DRESDEN**

Juristische Fakultät Fachrichtung Strafrecht, Institut für Kriminologie, Lehrstuhl für Kriminalprognose

ENTWICKLUNG EINES OPTIMALEN VERFAHRENS ZUR EROBERUNG DES GELDSPEICHERS IN ENTENHAUSEN

Mickey Mouse

Geboren am: 2. Januar 1990 in Dresden

DISSERTATION

zur Erlangung des akademischen Grades

DOKTOR DER RECHTE (DR. JUR.)

Erstgutachter

Dagobert Duck

Zweitgutachter

Mac Moneysac

Eingereicht am: 18. Februar 2015

Verteidigt am: 15. März 2015



ZUSAMMENFASSUNG

Dies hier ist ein Blindtext zum Testen von Textausgaben. Wer diesen Text liest, ist selbst schuld. Der Text gibt lediglich den Grauwert der Schrift an. Ist das wirklich so? Ist es gleichgültig, ob ich schreibe: „Dies ist ein Blindtext“ oder „Huardest gefburn“? Kjift – mitnichten! Ein Blindtext bietet mir wichtige Informationen. An ihm messe ich die Lesbarkeit einer Schrift, ihre Anmutung, wie harmonisch die Figuren zueinander stehen und prüfe, wie breit oder schmal sie läuft. Ein Blindtext sollte möglichst viele verschiedene Buchstaben enthalten und in der Originalsprache gesetzt sein. Er muss keinen Sinn ergeben, sollte aber lesbar sein. Fremdsprachige Texte wie „Lorem ipsum“ dienen nicht dem eigentlichen Zweck, da sie eine falsche Anmutung vermitteln.

ABSTRACT

Hello, here is some text without a meaning. This text should show what a printed text will look like at this place. If you read this text, you will get no information. Really? Is there no information? Is there a difference between this text and some nonsense like “Huardest gefburn”? Kjift – not at all! A blind text like this gives you information about the selected font, how the letters are written and an impression of the look. This text should contain all letters of the alphabet and it should be written in of the original language. There is no need for special content, but the length of words should match the language.

5.3 ABSCHLUSSARBEIT (KOLLABORATIV)

Alle zusätzlichen Angaben außerhalb des Argumentes von `\author` werden für beide Autoren gleichermaßen übernommen.⁵⁸ Die Angaben innerhalb des Argumentes von `\author` werden den jeweiligen, mit `\and` getrennten Autoren zugeordnet.⁵⁹ Ohne die Verwendung von `\and` kann natürlich auch nur ein Autor aufgeführt werden. Außerdem sei auf die Verwendung von `\subject` anstelle von `\thesis` mit einem speziellen Wert aus [Tabelle 2.1](#) hingewiesen.

```
\RequirePackage[ngerman=ngerman-x-latest]{hyphsubst}
\documentclass[ngerman,BCOR=4mm]{tudscrreprt}
\usepackage{selinput}\SelectInputMappings{adieresis={ä},germandbls={ß}}
\usepackage[T1]{fontenc}
\usepackage{babel}
\usepackage{isodate}
\begin{document}
\faculty{Juristische Fakultät}
\department{Fachrichtung Strafrecht}
\institute{Institut für Kriminologie}
\chair{Lehrstuhl für Kriminalprognose}
\date{18.02.2015}
\title{%
  Entwicklung eines optimalen Verfahrens zur Eroberung des
  Geldspeichers in Entenhausen
}
\subject{master}
\graduation[M.Sc.]{Master of Science}
\author{%
  Mickey Mouse
  \matriculationnumber{12345678}
  \dateofbirth{2.1.1990}
  \placeofbirth{Dresden}
\and%
  Donald Duck
  \matriculationnumber{87654321}
  \dateofbirth{1.2.1990}
  \placeofbirth{Berlin}
}
\matriculationyear{2010}
\supervisor{Dagobert Duck \and Mac Moneysac}
\professor{Prof. Dr. Kater Karlo}
\maketitle
\end{document}
```

⁵⁸In diesem Beispiel `\matriculationyear`.

⁵⁹In diesem Beispiel `\matriculationnumber`, `\dateofbirth` und `\placeofbirth`.



**TECHNISCHE
UNIVERSITÄT
DRESDEN**

Juristische Fakultät Fachrichtung Strafrecht, Institut für Kriminologie, Lehrstuhl für Kriminalprognose

Master-Arbeit

ENTWICKLUNG EINES OPTIMALEN VERFAHRENS ZUR EROBERUNG DES GELDSPEICHERS IN ENTENHAUSEN

Mickey Mouse

Geboren am: 2. Januar 1990 in Dresden
Matrikelnummer: 12345678
Immatrikulationsjahr: 2010

Donald Duck

Geboren am: 1. Februar 1990 in Berlin
Matrikelnummer: 87654321
Immatrikulationsjahr: 2010

zur Erlangung des akademischen Grades

Master of Science (M.Sc.)

Betreuer

**Dagobert Duck
Mac Moneysac**

Betreuender Hochschullehrer
Prof. Dr. Kater Karlo

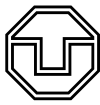
Eingereicht am: 18. Februar 2015

5.4 AUFGABENSTELLUNG (KOLLABORATIV)

Eine Aufgabenstellung für eine wissenschaftliche Arbeit ist mithilfe der Umgebung `task` oder dem Befehl `\taskform` aus dem Paket `tudscrsupervisor` folgendermaßen dargestellt werden.

```
\RequirePackage[ngerman=ngerman-x-latest]{hyphsubst}
\documentclass[ngerman]{tudscrreprt}
\usepackage{selinput}\SelectInputMappings{adieresis={ä},germandbls={ß}}
\usepackage[T1]{fontenc}
\usepackage{babel}
\usepackage{isodate}
\usepackage{tudscrsupervisor}
\usepackage{enumitem}\setlist{noitemsep}
\begin{document}
\faculty{Juristische Fakultät}\department{Fachrichtung Strafrecht}
\institute{Institut für Kriminologie}\chair{Lehrstuhl für Kriminalprognose}
\title{%
  Entwicklung eines optimalen Verfahrens zur Eroberung des
  Geldspeichers in Entenhausen
}
\thesis{master}\graduation[M.Sc.]{Master of Science}
\author{%
  Mickey Mouse\matriculationnumber{12345678}
  \dateofbirth{2.1.1990}\placeofbirth{Dresden}
  \course{Klinische Prognostik}\discipline{Individualprognose}
}
\and%
  Donald Duck\matriculationnumber{87654321}
  \dateofbirth{1.2.1990}\placeofbirth{Berlin}
  \course{Statistische Prognostik}\discipline{Makrosoziologische Prognosen}
}\matriculationyear{2010}\issuedate{1.4.2015}\duedate{1.10.2015}
\supervisor{Dagobert Duck \and Mac Moneysac}
\professor{Prof. Dr. Kater Karlo}\chairman{Prof. Dr. Primus von Quack}
\taskform[pagestyle=empty]{%
  Momentan ist das besagte Thema in aller Munde. Insbesondere wird es
  gerade in vielen--- wenn nicht sogar in allen--- Medien diskutiert.
  Es ist momentan noch nicht abzusehen, ob und wann sich diese Situation
  ändert. Eine kurzfristige Verlagerung aus dem Fokus der Öffentlichkeit
  wird nicht erwartet.

  Als Ziel dieser Arbeit soll identifiziert werden, warum das Thema
  gerade so omnipräsent ist und wie man diesen Effekt abschwächen
  könnte. Zusätzlich sollen Methoden entwickelt werden, wie sich ein
  ähnlicher Vorgang zukünftig vermeiden ließe.
}%
\item Recherche
\item Analyse
\item Entwicklung eines Konzeptes
\item Anwendung der entwickelten Methodik
\item Dokumentation und grafische Aufbereitung der Ergebnisse
}
\end{document}
```



AUFGABENSTELLUNG FÜR DIE ANFERTIGUNG EINER MASTER-ARBEIT

Studiengang:	Klinische Prognostik	Statistische Prognostik
Studienrichtung:	Individualprognose	Makrosoziologische Prognosen
Name:	Mickey Mouse	Donald Duck
Matrikelnummer:	12345678	87654321
Immatrikulationsjahr:	2010	2010
Titel:	Entwicklung eines optimalen Verfahrens zur Eroberung des Geldspeichers in Entenhausen	

ZIELE DER ARBEIT

Momentan ist das besagte Thema in aller Munde. Insbesondere wird es gerade in vielen – wenn nicht sogar in allen – Medien diskutiert. Es ist momentan noch nicht abzusehen, ob und wann sich diese Situation ändert. Eine kurzfristige Verlagerung aus dem Fokus der Öffentlichkeit wird nicht erwartet.

Als Ziel dieser Arbeit soll identifiziert werden, warum das Thema gerade so omnipräsent ist und wie man diesen Effekt abschwächen könnte. Zusätzlich sollen Methoden entwickelt werden, wie sich ein ähnlicher Vorgang zukünftig vermeiden ließe.

SCHWERPUNKTE DER ARBEIT

- Recherche
- Analyse
- Entwicklung eines Konzeptes
- Anwendung der entwickelten Methodik
- Dokumentation und grafische Aufbereitung der Ergebnisse

Betreuer:	Dagobert Duck Mac Moneysac
Ausgehändigt am:	1. April 2015
Einzureichen am:	1. Oktober 2015

Prof. Dr. Primus von Quack
Prüfungsausschussvorsitzender

Prof. Dr. Kater Karlo
Betreuender Hochschullehrer

5.5 GUTACHTEN

Nach dem Laden des Paketes `tudscrsupervisor` kann ein Gutachten für eine wissenschaftliche Arbeit mit der `evaluation`-Umgebung oder dem Befehl `\evaluationform` erstellt werden.

```
\RequirePackage[ngerman=ngerman-x-latest]{hyphsubst}
\documentclass[ngerman]{tudscrreprt}
\usepackage{selinput}\SelectInputMappings{adieresis={ä},germandbls={ß}}
\usepackage[T1]{fontenc}
\usepackage{babel}
\usepackage{isodate}
\usepackage{tudscrsupervisor}
\usepackage{enumitem}\setlist{noitemsep}
\begin{document}
\faculty{Juristische Fakultät}\department{Fachrichtung Strafrecht}
\institute{Institut für Kriminologie}\chair{Lehrstuhl für Kriminalprognose}
\date{18.02.2015}
\author{Mickey Mouse}
\title{%
  Entwicklung eines optimalen Verfahrens zur Eroberung des
  Geldspeichers in Entenhausen
}
\thesis{diploma}
\graduation[Dipl.-Ing.]{Diplomingenieur}
\matriculationnumber{12345678}
\matriculationyear{2010}
\dateofbirth{2.1.1990}
\placeofbirth{Dresden}
\referee{Dagobert Duck}
\evaluationform[pagestyle=empty]{%
  Als Ziel dieser Arbeit sollte identifiziert werden, warum das Thema
  gerade so omnipräsent ist und wie man diesen Effekt abschwächen
  könnte. Zusätzlich sollten Methoden entwickelt werden, wie sich ein
  ähnlicher Vorgang zukünftig vermeiden ließe.
}{%
  Die Arbeitet gliedert sich in mehrere Kapitel auf unzähligen Seiten. Im den
  ersten beiden Kapiteln wird dies und das besprochen. Im darauffolgenden auch
  noch jenes. Im vorletzten Kapitel wird alles betrachtet und im letzten
  erfolgt die Zusammenfassung und der Ausblick.
}{%
  Die dargestellten Ergebnisse der vorliegenden Arbeit genügen den in der
  Aufgabenstellung formulierten Arbeitsschwerpunkten. Sowohl Form, Ausdruck
  und Terminologie als auch Struktur befriedigen die Anforderungen an eine
  wissenschaftliche Arbeit. Die einzelnen Kapitel können hinsichtlich ihrer
  inhaltlichen Tiefe in Relation zu dem dazu benötigten Umfang als ausgewogen
  betrachtet werden. Die verwendeten Grafiken und Tabellen sind sehr gut
  aufbereitet und tragen insgesamt zum besseren Verständnis bei. Fachliche
  Begrifflichkeiten werden sehr sauber verwendet, orthografische Mängel sind
  keine zu finden.
}{1,0 (sehr gut)}
\end{document}
```



GUTACHTEN FÜR DIE DIPLOMARBEIT

Name: Mickey Mouse
Matrikelnummer: 12345678
Immatrikulationsjahr: 2010
Titel: Entwicklung eines optimalen Verfahrens zur Eroberung des
Geldspeichers in Entenhausen

AUFGABENSTELLUNG

Als Ziel dieser Arbeit sollte identifiziert werden, warum das Thema gerade so omnipräsent ist und wie man diesen Effekt abschwächen könnte. Zusätzlich sollten Methoden entwickelt werden, wie sich ein ähnlicher Vorgang zukünftig vermeiden ließe.

INHALT

Die Arbeit gliedert sich in mehrere Kapitel auf unzähligen Seiten. Im den ersten beiden Kapiteln wird dies und das besprochen. Im darauffolgenden auch noch jenes. Im vorletzten Kapitel wird alles betrachtet und im letzten erfolgt die Zusammenfassung und der Ausblick.

BEWERTUNG

Die dargestellten Ergebnisse der vorliegenden Arbeit genügen den in der Aufgabenstellung formulierten Arbeitsschwerpunkten. Sowohl Form, Ausdruck und Terminologie als auch Struktur befriedigen die Anforderungen an eine wissenschaftliche Arbeit. Die einzelnen Kapitel können hinsichtlich ihrer inhaltlichen Tiefe in Relation zu dem dazu benötigten Umfang als ausgewogen betrachtet werden. Die verwendeten Grafiken und Tabellen sind sehr gut aufbereitet und tragen insgesamt zum besseren Verständnis bei. Fachliche Begrifflichkeiten werden sehr sauber verwendet, orthografische Mängel sind keine zu finden.

Die Arbeit wird mit der Note **1,0 (sehr gut)** bewertet.

Dresden, 18. Februar 2015

Dagobert Duck
Gutachter

5.6 AUSHANG

Das Paket `tudscrsupervisor` stellt die Umgebung `notice` für das Anfertigen allgemeiner Aushänge sowie den Befehl `\noticeform` für die Ausschreibung wissenschaftlicher Arbeiten bereit.

```
\RequirePackage[ngerman=ngerman-x-latest]{hyphsubst}
\documentclass[ngerman]{tudscrreprt}
\usepackage{selinput}\SelectInputMappings{adieresis={ä},germandbls={ß}}
\usepackage[T1]{fontenc}
\usepackage{babel}
\usepackage{isodate}
\usepackage{tudscrsupervisor}
\usepackage{enumitem}\setlist{noitemsep}
\begin{document}
\faculty{Juristische Fakultät}\department{Fachrichtung Strafrecht}
\institute{Institut für Kriminologie}\chair{Lehrstuhl für Kriminalprognose}
\date{18.02.2015}
\title{%
  Entwicklung eines optimalen Verfahrens zur Eroberung des
  Geldspeichers in Entenhausen
}
\contactperson{%
  Dagobert Duck\office{Dingens-Bau, Zimmer~08}
  \emailaddress{dagobert.duck@tu-dresden.de}\telephone{+49 351 463-12345}
\and%
  Mac Moneysac\office{Dingens-Bau, Zimmer~15}
  \emailaddress{mac.moneysac@tu-dresden.de}\telephone{+49 351 463-54321}
}
\noticeform[Angebot für eine Studien-/Diplomarbeit,pagestyle=empty]{%
  Momentan ist das besagte Thema in aller Munde. Insbesondere wird es gerade
  in vielen--- wenn nicht sogar in allen--- Medien diskutiert. Es ist momentan
  nicht abzusehen, ob und wann sich diese Situation ändert. Eine kurzfristige
  Verlagerung aus dem Fokus der Öffentlichkeit wird nicht erwartet.

  Als Ziel dieser Arbeit soll identifiziert werden, warum das Thema gerade so
  omnipräsent ist und wie man diesen Effekt abschwächen könnte. Zusätzlich
  sollen Methoden entwickelt werden, wie sich ein ähnlicher Vorgang zukünftig
  vermeiden ließe.
  \begin{center}
\medskip\includegraphics[width=.7\linewidth]{DDC-21}
\renewcommand*{\figureformat}{\figurename}
\captionof{figure}{Thematisch passendes Bild}
\end{center}
}{%
  \item Recherche
  \item Analyse
  \item Entwicklung eines Konzeptes
  \item Anwendung der entwickelten Methodik
  \item Dokumentation und grafische Aufbereitung der Ergebnisse
}
\end{document}
```



18. Februar 2015

**ANGEBOT FÜR EINE STUDIEN-/DIPLOMARBEIT
ENTWICKLUNG EINES OPTIMALEN VERFAHRENS ZUR EROBERUNG DES GELDSPEICHERS
IN ENTENHAUSEN**

Momentan ist das besagte Thema in aller Munde. Insbesondere wird es gerade in vielen – wenn nicht sogar in allen – Medien diskutiert. Es ist momentan nicht abzusehen, ob und wann sich diese Situation ändert. Eine kurzfristige Verlagerung aus dem Fokus der Öffentlichkeit wird nicht erwartet.

Als Ziel dieser Arbeit soll identifiziert werden, warum das Thema gerade so omnipräsent ist und wie man diesen Effekt abschwächen könnte. Zusätzlich sollen Methoden entwickelt werden, wie sich ein ähnlicher Vorgang zukünftig vermeiden ließe.

DRESDEN
concept

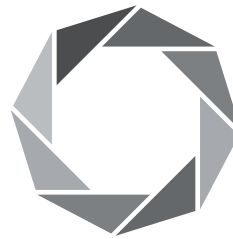


Abbildung: Thematisch passendes Bild

SCHWERPUNKTE DER ARBEIT

- Recherche
- Analyse
- Entwicklung eines Konzeptes
- Anwendung der entwickelten Methodik
- Dokumentation und grafische Aufbereitung der Ergebnisse

ANSPRECHPARTNER

Dagobert Duck
Dingens-Bau, Zimmer 08
+49 351 463-12345
dagobert.duck@tu-dresden.de

Mac Moneysac
Dingens-Bau, Zimmer 15
+49 351 463-54321
mac.moneysac@tu-dresden.de

5.7 VORLAGE FÜR EINE WISSENSCHAFTLICHEN ARBEIT

Die meisten Anwender der TUD-KOMA-Script-Klassen sind Studenten oder angehörige der Technischen Universität Dresden, die ihre ersten Schritte mit $\text{\LaTeX} 2_{\epsilon}$ beim Verfassen einer wissenschaftlichen Arbeit oder ähnlichem machen. Während der Einstiegsphase in $\text{\LaTeX} 2_{\epsilon}$ kann ein Anfänger sehr schnell aufgrund der großen Anzahl an empfohlenen Pakete sowie der teilweise diametral zueinander stehenden Hinweise überfordert sein. Mit dem Tutorial [treatise.pdf](#) soll versucht werden, ein wenig Licht ins Dunkel zu bringen. Es erhebt jedoch keinerlei Anspruch, vollständig oder perfekt zu sein. Einige der darin vorgestellten Möglichkeiten lassen sich mit Sicherheit auch anders, einfacher und/oder besser lösen. Dennoch ist es gerade für Neulinge – vielleicht auch für den einen oder anderen \LaTeX -Veteran – als Leitfaden für die Erstellung einer wissenschaftlichen Arbeit gedacht.

5.8 EIN BEITRAG ZUM MATHEMATISCHEN SATZ IN $\text{\LaTeX} 2_{\epsilon}$

Das Tutorial [mathtype.pdf](#) richtet sich an alle Anwender, die in ihrem \LaTeX -Dokument mathematische Formeln setzen wollen. In diesem wird ausführlich darauf eingegangen, wie mit wenigen Handgriffen ein typografisch sauberer Mathematiksatz zu bewerkstelligen ist.

5.9 ÄNDERUNG DER TRENNZEICHEN IM MATHEMATIKMODUS

Sollen beim Verfassen eines \LaTeX -Dokumentes Daten in einem Zahlenformat importiert werden, welches nicht den Gepflogenheiten der Dokumentsprache entspricht, kommt es meist zu unschönen Ergebnissen bei der Ausgabe. Einfachstes Beispiel sind Daten, in denen als Dezimaltrennzeichen ein Punkt verwendet wird, wie es im englischsprachigen Raum der Fall ist. Sollen diese in einem Dokument deutscher Sprache eingebunden werden, müssten diese normalerweise allesamt angepasst und das ursprüngliche Dezimaltrennzeichen durch ein Komma ersetzt werden. Dieser Schritt wird mit dem TUD-KOMA-Script-Paket [mathswap](#) automatisiert. Wie dies genau funktioniert, wird im Tutorial [mathswap.pdf](#) erläutert.

KAPITEL 6

UNERLÄSSLICHE UND BEACHTENSWERTE PAKETE

6.1 VON DEN NEUEN HAUPTKLASSEN BENÖTIGTE PAKETE

6.1.1 ERFORDERLICHE PAKETE BEI DER SCHRIFTINSTALLATION

Für die Installation der Schriften sind die folgend genannten Pakete von *essentieller* Bedeutung und daher *zwingend* notwendig. Das Vorhandensein dieser wird durch die jeweiligen Schriftinstallationskripte (siehe [Unterkapitel 1.2](#)) geprüft und die Installation beim Fehlen eines oder mehrerer Pakete mit einer entsprechenden Warnung abgebrochen.

fontinst (fontware)

Dieses Paket wird für die Installation der Schriften **Univers** sowie **DIN BOLD** benötigt. Weiterhin ist [fontware](#) der L^AT_EX-Distribution für die Schriftkonvertierung notwendig.

cmbright, hfbright, cm-super

Alle mathematischen Glyphen und Symbole, die nicht in den **Univers**-Schriften enthalten sind, werden diesem Paket entnommen. Außerdem werden für die T1-Schriftkodierung die beiden Pakete [cm-super](#) und [hfbright](#) benötigt. Aus dem Paket [lmodern](#) werden zusätzlich die Schreibmaschinenschriften verwendet.

iwona

Sowohl für **Univers** als auch für **DIN BOLD** werden fehlenden Glyphen entnommen.

6.1.2 NOTWENDIGE PAKETE FÜR DIE VERWENDUNG DER HAUPTKLASSEN

Diese Pakete werden von den neuen Klassen zwingend benötigt und automatisch geladen.

v2.02

koma-script, typearea, scrlayer-scrpage, scrbase, scrwfile

Das KOMA-Script-Bundle ist die zentrale Grundlage für TUD-KOMA-Script. Neben den Klassen [scrbook](#), [scrreprt](#) und [scrartcl](#) wird das Paket [scrbase](#) benötigt. Dieses erlaubt das Definieren von Klassenoptionen im Stil von KOMA-Script, welche auch noch nach dem Laden der Klasse mit den Befehlen [\TUOption](#) und [\TUOptions](#) geändert werden können. Für die Bereitstellung der [tudheadings](#)-Seitenstile ist das Paket [scrlayer-scrpage](#) notwendig. Wenn es nicht durch den Anwender – mit beliebigen Optionen – geladen wird, erfolgt dies am Ende der Präambel automatisch durch TUD-KOMA-Script.

kvsetkeys

Hiermit wird das von [scrbase](#) geladene Paket [keyval](#) verbessert, welches das Definieren von Klassen- und Paketoptionen sowie Parametern nach dem Schlüssel-Wert-Prinzip ermöglicht. Das Verhalten für unbekannte Schlüssel kann mit [kvsetkeys](#) festgelegt werden.

etoolbox

Es werden viele Funktionen zum Testen und zur Ablaufkontrolle bereitgestellt und das einfache Manipulieren vorhandener Makros ermöglicht.

geometry

Das Paket wird zum Festlegen der Seitenränder respektive des Satzspiegels verwendet. Ein Weiterreichen zusätzlicher Optionen an das Paket wird dringlich nicht empfohlen.

textcase

Mit [\MakeTextUppercase](#) wird die Großschreibung der Überschriften in **DIN BOLD** erzwungen. Im *Ausnahmefall* kann dies mit [\NoCaseChange](#) unterbunden werden.

graphicx

Dies ist das De-facto-Standard-Paket zum Einbinden von Grafiken. Zum Setzen des Logos der Technischen Universität Dresden im Kopf wird [\includegraphics](#) genutzt. Es kann auch durch den Anwender in der Präambel geladen werden.

xcolor

Damit werden die Farben des Corporate Designs zur Verwendung im Dokument definiert. Genauer ist bei der Beschreibung von `tudscrcolor` in [Unterkapitel 4.2](#) zu finden. Ein Laden beider Pakete in der Präambel durch den Nutzer ist problemlos möglich.

environ

Es wird eine verbesserte Deklaration von Umgebungen ermöglicht, bei der auch beim Abschluss der Umgebung auf die übergebenen Parameter zugegriffen werden kann. Dies wird die Neugestaltung der `abstract`-Umgebung benötigt.

trimspaces

Bei mehreren Eingabefeldern für den Anwender werden die Argumente mithilfe dieses Paketes um eventuell angegebene, unnötige Leerzeichen befreit.

Möchten Sie eines der hier aufgezählten Pakete selber nutzen, es jedoch mit bestimmten Optionen laden, so sollten diese bereits *vor* der Definition der Dokumentklasse an das Paket weitergereicht werden, falls bei der Beschreibung nichts anderweitiges angegeben worden ist.

Beispiel: Das Weiterreichen von Optionen an Pakete muss folgendermaßen erfolgen:

```
\PassOptionsToPackage{<Optionenliste>}{<Paket>}  
\documentclass[<Klassenoptionen>]{tudscr...}
```

6.2 DURCH TUD-KOMA-SCRIPT DIREKT UNTERSTÜTZE PAKETE

Wird eines der genannten Pakete geladen, so wird die Funktionalität von TUD-KOMA-Script durch dieses verbessert beziehungsweise erweitert oder es wird ein Bugfix bereitgestellt.

hyperref

Hiermit können in einem PDF-Dokument Lesezeichen, Querverweise und Hyperlinks erstellt werden. Wird es geladen, sind außerdem die Option `tudbookmarks` sowie der Befehl `\tudbookmark` nutzbar. Das Paket `bookmark` erweitert die Unterstützung nochmals. Beide genannten Pakete sollten – bis auf sehr wenige Ausnahmen wie beispielsweise `glossaries` – als letztes in der Präambel eingebunden werden.

isodate

Dieses Paket formatiert mit `\printdate{<Datum>}` die Ausgabe eines Datums automatisch in ein spezifiziertes Format. Wird es geladen, werden alle Datumsfelder, welche durch die TUD-KOMA-Script-Klassen definiert wurden,⁶⁰ in diesem Format ausgegeben.

multicol

Hiermit kann jeglicher beliebiger Inhalt in zwei oder mehr Spalten ausgegeben werden, wobei – im Gegensatz zur KOMA-Script-Option `twocolumn` – für einen Spaltenausgleich gesorgt wird. Unterstützt wird das Paket innerhalb der Umgebungen `abstract` und `tudpage`.

quoting

\LaTeX bietet von Haus aus *zwei* verschiedene Umgebungen für Zitate und ähnliches. Beide sind in ihrer Ausprägung starr und ignorieren beispielsweise die Einstellungen der KOMA-Script-Option `parskip`. Dies wird durch die Umgebung `quoting` verbessert. Wird das Paket geladen, kommt diese innerhalb der `abstract`-Umgebung zum Einsatz.

ragged2e

Das Paket verbessert den Flattersatz, indem für diesen die Worttrennung aktiviert wird.

⁶⁰`\date`, `\dateofbirth` und `\defensedate` sowie aus `tudscrsupervisor` `\duedate` und `\issuedate`

6.3 EMPFEHLENSWERTE PAKETE

In diesem Unterkapitel wird eine Vielzahl an Paketen – zumeist kurz – vorgestellt, welche sich für mich persönlich bei der Arbeit mit \LaTeX bewährt haben. Einige davon werden außerdem im Tutorial [treatise.pdf](#) in ihrer Anwendung beschrieben. Für detaillierte Informationen sowie bei Fragen zu den einzelnen Paketen sollte die jeweilige Dokumentation zu Rate gezogen werden,⁶¹ das Lesen der hier gegebenen Kurzbeschreibung ersetzt dies in keinem Fall.

6.3.1 PAKETE ZUR VERWENDUNG IN JEDEM DOKUMENT

Die hier vorgestellten Pakete gehören meiner Meinung nach in die Präambel eines jeden Dokumentes. Egal, in welcher Sprache das Dokument verfasst wird, sollte diese mit dem Paket [babel](#) definiert werden – auch wenn dies Englisch ist. Für deutschsprachige Dokumente ist für eine annehmbare Worttrennung das Paket [hyphsubst](#) unbedingt zu verwenden.

fontenc

Das Paket erlaubt Festlegung der Zeichensatzkodierung des Ausgabefonts. Als Voreinstellung ist die Ausgabe als 7-bit kodierte Schrift gewählt, was unter anderem dazu führt, dass keine echten Umlaute im erzeugten PDF-Dokument verwendet werden. Um auf 8-bit-Schriften zu schalten, sollte man `\usepackage[T1]{fontenc}` nutzen.

selinput, inputenc

Hiermit erfolgt die (automatische) Festlegung der Eingabekodierung. Diese ist vom genutzten [Editors](#) ([Unterkapitel 7.1](#)) und den darin gewählten Einstellung abhängig. Mit:

```
\usepackage{selinput}
\SelectInputMappings{adieresis={ä},germandbls={ß}}
```

wird es verwendet. Dies macht den Quelltext portabel, womit beispielsweise einfach via Copy & Paste ein [Minimalbeispiel](#)⁶² bei Problemstellungen in einem Forum bereitgestellt werden kann. Alternativ dazu lässt sich mit dem Paket [inputenc](#) die zu verwendende Eingabekodierung manuell einstellen (`\usepackage[<Eingabekodierung>]{inputenc}`).

babel, polyglossia

Mit diesem Paket erfolgt die Einstellung der im Dokument verwendeten Sprache(n). Bei mehreren angegebenen Sprachen ist die zuletzt geladene die Hauptsprache des Dokumentes. Die gewünschten Sprachen sollten als nicht als Paketooption sondern als Klassenoption und gesetzt werden, damit auch andere Pakete auf die Spracheinstellungen zugreifen können. Für deutschsprachige Dokumente ist die Option `ngerman` für die neue oder `german` für die alte deutsche Rechtschreibung zu verwenden.

Mit dem Laden von [babel](#) und der dazugehörigen Sprachen werden sowohl die Trennmuster als auch die sprachabhängigen Bezeichner angepasst. Von einer Verwendung der obsoleten Pakete [german](#) beziehungsweise [ngerman](#) anstelle von [babel](#) wird abgeraten. Für \LuaTeX und \XeTeX kann das Paket [polyglossia](#) genutzt werden.

microtype

Dieses Paket kümmert sich um den optischen Randausgleich⁶³ und das Nivellieren der Wortzwischenräume⁶⁴ im Dokument. Es funktioniert nicht mit der klassischen \TeX -Engine, wohl jedoch mit \pdfTeX als auch \LuaTeX sowie \XeTeX .

⁶¹Kommandozeile/Terminal: `texdoc <Paketname>`

⁶²<http://www.komascript.de/minimalbeispiel>

⁶³englisch: protrusion, margin kerning

⁶⁴englisch: font expansion

hyphsubst, dehyph-exptl

Die möglichen Trennstellen von Wörtern wird von \LaTeX mithilfe eines Algorithmus berechnet. Dieser wird für deutschsprachige Texte mit dem Paket **hyphsubst** entscheidend verbessert. Es muss bereits *vor* der Dokumentklasse wie folgt geladen werden:

```
\RequirePackage[ngerman=ngerman-x-latest]{hyphsubst}
```

In [Unterkapitel 7.3](#) wird genauer auf das Zusammenspiel von **hyphsubst** und **babel** sowie **fontenc** eingegangen, ein Blick dahin wird dringend empfohlen. Zusätzliche werden dort weitere Hinweise für eine verbesserte Worttrennung gegeben.

6.3.2 PAKETE ZUR SITUATIVEN VERWENDUNG

Die nachfolgenden Pakete sollten nicht zwangsweise in jedem Dokument geladen werden sondern nur, falls dies auch tatsächlich notwendig ist. Zur besseren Übersicht wurde versucht, diese thematisch passend zu gruppieren. Daraus lässt sich keinerlei Wertung bezüglich ihrer Nützlichkeit oder meiner persönlichen Wertschätzung ableiten.

TYPOGRAFIE UND LAYOUT

setspace

Die Vergrößerung des Zeilenabstandes wird:

- I. viel zu häufig und völlig unnötig gefordert und
- II. schließlich auch noch zu groß gewählt.

Die Forderung nach Erhöhung des Zeilenabstandes – in der Typografie als Durchschuss bezeichnet – kommt noch aus den Zeiten der Textverarbeitung mit der Schreibmaschine. Ein einzeliger Zeilenabstand bedeutete hier, dass die Unterlängen der oberen Zeile genau auf der Höhe der Oberlängen der folgenden Zeile lagen. Ein anderthalbzeiliger Zeilenabstand erzielte hier somit einen akzeptablen Durchschuss. Eine Erhöhung des Durchschusses bei der Verwendung von \LaTeX ist an und für sich nicht notwendig. Sinnvoll ist dies nur, wenn im Fließtext serifenlose Schriften zum Einsatz kommen, um die damit verbundene schlechte Lesbarkeit etwas zu verbessern.

Ist die Erhöhung des Durchschusses wirklich notwendig, sollte das Paket **setspace** genutzt werden. Dieses stellt den Befehl `\setstretch{<Faktor>}` zur Verfügung, mit dem der Durchschuss respektive Zeilenabstand angepasst werden kann. Der Wert des Faktors ist standardmäßig auf 1 gestellt und sollte maximal bis 1.25 vergrößert werden. Der Befehl `\onehalfspacing` aus diesem Paket setzt diesen Wert auf eben genau 1.25. Allerdings ist hier anzumerken, dass die Vergrößerung des Zeilenabstandes – so wie ich es mir angelesen habe – aus der Sicht eines Typographen keine Spielerei ist sondern vielmehr allein der Lesbarkeit des Textes dient und möglichst gering ausfallen sollte.

Ziel ist es, beim Lesen nach dem Beenden der aktuellen Zeile das Auffinden der neuen Zeile zu vereinfachen. Bei Serifen ist dies durch die Betonung der Grundlinie sehr gut möglich. Bei serifenlosen Schriften – wie der im Corporate Design der Technischen Universität Dresden verwendeten **Univers** – ist dies schwieriger und ein erweiterter Abstand der Zeilen kann dabei durchaus hilfreich sein. Jedoch sollte nicht nach dem Motto „viel hilft viel“ verfahren werden. In diesem Dokument wurde als Faktor für den Zeilenabstand `\setstretch{1.1}` gewählt. Nach einer Einstellung des Zeilenabstandes sollte der Satzspiegel unbedingt mit `\recalctypearea` neu berechnet werden. Weitere Tipps sind in [Unterkapitel 7.7](#) sowie [Unterkapitel 7.8](#) zu finden.

csquotes

Das Paket stellt unter anderem den Befehl `\enquote{<Zitat>}` zur Verfügung, welcher Anführungszeichen in Abhängigkeit der gewählten Sprache setzt. Zusätzlich werden weitere Kommandos und Optionen für die spezifischen Anforderungen des Zitierens bei wissenschaftlichen Arbeiten angeboten. Außerdem wird es durch **biblatex** unterstützt und sollte zumindest bei dessen Verwendung geladen werden.

noindentafter

Mit diesem Paket lassen sich automatische Absatzeinzüge für selbst zu bestimmende Befehle und Umgebungen unterdrücken.

xspace

Mit **xspace** kann bei der Definition eigener Makros der Befehl `\xspace` genutzt werden. Dieser setzt ein gegebenenfalls notwendiges Leerzeichen automatisch. In [Unterkapitel 7.14](#) ist die Definition eines solchen Befehls exemplarisch ausgeführt.

ellipsis

In \LaTeX folgen den Befehlen für Auslassungspunkte (`\dots` und `\text{ellipsis}`) *immer* ein Leerzeichen. Dies kann unter Umständen unerwünscht sein. Mit dem Paket **ellipsis** wird das nachfolgende Leerzeichen – im Gegensatz zum Standardverhalten – nur gesetzt, wenn ein Satzzeichen und kein Buchstabe folgt, *siehe auch* [Unterkapitel 7.15](#).

xpunctuate

Die Funktionalität von **xspace** wird um die Beachtung von Interpunktionen erweitert.

DeLig

Hierbei handelt es sich um ein Java-Script, welches anhand eines Wörterbuches falsche Ligaturen innerhalb eines Dokumentes automatisiert entfernt. Wird **Univers** verwendet ist dies jedoch nicht notwendig, da diese keinerlei Ligaturen enthält, die insbesondere in deutschen Texten für einen guten Satz manuell aufgelöst werden müssten.⁶⁵ Mit \LuaTeX als Dokumentprozessor kann alternativ dazu auch **selnolig** verwendet werden.

balance

Dieses Paket ermöglicht einen Spaltenausgleich im Zweispaltensatz auf der letzten Dokumentseite. Alternativ dazu kann auch **multicol** verwendet werden.

RECHTSCHREIBUNG

Für die Rechtschreibkontrolle zeichnet im Normalfall der verwendete Editor verantwortlich. Dennoch gibt es einige wenige Pakete, welche sich diesem Thema widmen. Diese sind jedoch lediglich nutzbar, wenn \LuaTeX als Dokumentprozessor genutzt wird.

lua-check-hyphen

Hiermit lassen sich mit \LuaTeX Trennstellen am Zeilenende zur Prüfung markieren. Zum Thema der *korrekten Worttrennung* sei außerdem auf [Unterkapitel 7.3](#) verwiesen.

spelling

Wird \LuaTeX als Prozessor verwendet, wird mit diesem Paket der reine Textanteil aus dem \LaTeX -Dokument extrahiert – wobei Makros und aktive Zeichen entfernt werden – und in eine separate Textdatei geschrieben. Anschließend kann diese Datei mit einer externen Software zur Rechtschreibprüfung wie **GNU Aspell** oder **Hunspell** analysiert werden. Wird durch dieses Programm eine Liste falsch geschriebener Wörter ausgegeben, können diese mit **spelling** im PDF-Dokument hervorgehoben werden.

⁶⁵Das sind ff, fi, fl, ffi, und ffl bei den \LaTeX -Standardschriften.

SCHRIFTEN UND SONDERZEICHEN

lmodern

Soll mit den klassischen \LaTeX -Standardschriften gearbeitet werden, empfiehlt sich die Verwendung des Paketes **lmodern**. Dieses verbessert die Darstellung der Computer Modern sowohl am Bildschirm als auch beim finalen Druck.

cfr-lm

Dieses experimentelle Paket liefert weitere Schriftschnitte für das Paket **lmodern**.

newtx, newtxmath

Es werden einige alternative Schriften sowohl für den Fließtext (*Times* und *Helvetica*) als auch den Mathematikmodus bereitgestellt.

libertine

Das Paket stellt die Schriften Linux Libertine und Linux Biolinum zur Verfügung. Um diese Schriftart auch für den Mathematikmodus verwenden zu können, sollte **newtxmath** aus dem **newtx**-Bundle mit `\usepackage[libertine]{newtxmath}` in der Präambel eingebunden werden. Das Paket **libgreek** enthält griechische Buchstaben für Linux Libertine.

mweights

In $\text{\LaTeX} 2_{\epsilon}$ existieren die drei Schriftfamilien für Serifenschriften (`\rmfamily`), serifenlose Schriften (`\sffamily`) sowie die Schreibmaschinenschriften (`\ttfamily`). Deren Schriftstärke wird für gewöhnlich mit den beiden Befehlen `\mddefault` und `\bfdefault` einheitlich festgelegt. Bei der Verwendung unterschiedlicher Schriftpakete kann es unter Umständen zu Problemen bei den Schriftstärken kommen. Das Paket **mweights** erlaubt die individuelle Definition der Schriftstärke für jede der drei Schriftfamilien.

v2.02

fontspec

Wird als Dokumentprozessor nicht $\text{pdf}\text{\LaTeX}$ sondern $\text{Xe}\text{\LaTeX}$ oder $\text{Lua}\text{\LaTeX}$ verwendet, können mit diesem Paket Systemschriften im OpenType-Format eingebunden werden, womit sich die Auswahl der verwendbaren Schriften in einem \LaTeX -Dokument stark erweitert. Das Paket wird durch TUD-KOMA-Script unterstützt.

relsize

Die Größe einer Textauszeichnung kann relativ zur aktuellen Schriftgröße gesetzt werden.

textcomp

Es werden zusätzliche Symbole und Sonderzeichen wie beispielsweise das Promille- oder Eurozeichen sowie Pfeile für den Fließtext zur Verfügung gestellt.

Auch für (serifenlose) Mathematikschriften gibt es einige nützliche Pakete. Werden die Schriften des Corporate Designs genutzt, sei auf die Option **cdmath** verwiesen.

sansmathfonts, sansmath

Sollten die normalen \LaTeX -Schriften Computer Modern verwendet werden, kann man dieses Paket zum serifenlosen Setzen mathematischer Ausdrücke nutzen. Ein alternatives Paket mit der gleichen Zielstellung ist **sansmath**

sfmath

Dieses Paket verfolgt ein ähnliches Ziel, kann jedoch im Gegensatz zu **sansmath** nicht nur für Computer Modern sondern mit der entsprechenden Option auch für Latin Modern, Helvetica und Computer Modern Bright verwendet werden.

mathastext

Mit dem Paket wird das Ziel verfolgt, aus der genutzten Schrift für den Fließtext alle notwendigen Zeichen für den Mathematiksatz zu extrahieren.

MATHEMATIKSATZ

Dies sind Pakete, die Umgebungen und Befehle für den Mathematiksatz sowie das Setzen von Einheiten und Zahlen im Allgemeinen anbieten.

mathtools, amsmath

Dieses Paket stellt für das De-facto-Standard-Paket [amsmath](#) für Mathematikumgebungen Bugfixes zur Verfügung und erweitert dieses.

bm

Das Paket bietet mit `\bm` eine Alternative zu `\boldsymbol` im [Mathematiksatz](#)⁶⁶.

Für das typografisch korrekte Setzen von Einheiten – ein halbes Leerzeichen zwischen Zahl und *aufrecht* gesetzter Einheit – gibt es zwei gut nutzbare Pakete.

units

Dies ist ein einfaches und sehr zweckdienliches Paket zum Setzen von Einheiten und für die meisten Anforderungen völlig ausreichend.

siunitx

Dieses Paket ist in seinem Umfang im Vergleich deutlich erweitert. Neben Einheiten können zusätzlich auch Zahlen typografisch korrekt gesetzt werden. Die Ausgabe lässt sich in vielerlei Hinsicht an individuelle Bedürfnisse anpassen. Für deutschsprachige Dokumenten sollte die Lokalisierung angegeben werden. Mehr dazu in [Unterkapitel 7.12](#).

Die korrekte Formatierung von Zahlen ist häufig ein Problem bei der Verwendung von \LaTeX . Insbesondere, wenn in einem deutschsprachigen Dokument Daten im englischsprachigen Format verwendet werden, kommt es zu Problemen. Dafür wird im TUD-KOMA-Script-Bundle das Paket [mathswap](#) bereitgestellt. Dennoch gibt es zu diesem auch Alternativen.

icomma

Wird im Mathematikmodus nach dem Komma ein Leerzeichen gesetzt, wird dies bei der Ausgabe beachtet. Der Verfasser muss sich demzufolge jederzeit selbst um die typografisch korrekte Ausgabe kümmern.

ziffer

Für deutschsprachige Dokumente wird das Komma als Dezimaltrennzeichen zwischen zwei Ziffern definiert. Folgt dem Komma keine Ziffer, wird jederzeit der obligatorische Freiraum gesetzt, was meiner Meinung nach besser als das Verhalten von [icomma](#) ist.

ionumbers

Dieses Paket ist mir tatsächlich erst bei der Arbeit an [mathswap](#) bekannt geworden. Es bietet mehr Funktionalitäten und kann als Alternative dazu betrachtet werden.

Weitere Hinweise und Anwendungsfälle zur mathematischen Typografie werden in [Unterkapitel 5.8](#) sowie [Unterkapitel 5.9](#) gegeben.

VERZEICHNISSE ALLER ART

Neben dem Erstellen des eigentlichen Dokumentes sind für eine wissenschaftliche Arbeit meist auch allerhand Verzeichnisse gefordert. Fester Bestandteil ist dabei das Literaturverzeichnis, auch ein Abkürzungs- und Formelzeichen- beziehungsweise Symbolverzeichnis werden häufig gefordert. Gegebenenfalls wird auch noch ein Glossar benötigt. Hier werden die passenden

⁶⁶<http://tex.stackexchange.com/questions/3238>

Pakete vorgestellt. Sollen im Dokument komplette Quelltexte oder auch nur Auszüge daraus erscheinen und für diese auch gleich ein entsprechendes Verzeichnis generiert werden, so sei auf das Paket **listings** in [Unterkapitel 6.3.2 auf Seite 100](#) verwiesen.

biblatex

Das Paket kann als legitimer Nachfolger zu BibTeX gesehen werden. Ähnlich dazu bietet **biblatex** die Möglichkeit, Literaturdatenbanken einzubinden und verschiedene Stile der Referenzierung und Darstellung des Literaturverzeichnisses auszuwählen.

Mit **biblatex** ist die Anpassung eines bestimmten Stiles wesentlich besser umsetzbar als mit BibTeX. Wird **biber** für die Sortierung des Literaturverzeichnisses genutzt, ist die Verwendung einer UTF-8-kodierten Literaturdatenbank problemlos möglich. In Verbindung mit **biblatex** wird die zusätzliche Nutzung des Paketes **csquotes** sehr empfohlen.

acro, acronym

Soll lediglich ein Abkürzungsverzeichnis erstellt werden, ist dieses Paket die erste Wahl. Es stellt Befehle zur Definition von Abkürzungen sowie zu deren Verwendung im Text und zur sortierten Ausgabe eines Verzeichnisses bereit. Alternativ dazu kann das Paket **acronym** verwendet werden. Die Sortierung des Abkürzungsverzeichnisses muss hier allerdings manuell durch den Anwender erfolgen.

glossaries, nomencl

Dies ist ein sehr mächtiges Paket zum Erstellen eines Glossars sowie Abkürzungs- und Symbolverzeichnisses. Die mannigfaltige Anzahl an Optionen ist zu Beginn eventuell etwas abschreckend. Insbesondere wenn Verzeichnisse für Abkürzungen *und* Formelzeichen beziehungsweise Symbole benötigt werden, sollte man dieses Paket in Erwägung ziehen.

Alternativ dazu kann für ein Symbolverzeichnis auch lediglich eine manuell gesetzte Tabelle genutzt werden. Das hierfür sehr häufig empfohlene Paket **nomencl** bietet meiner Meinung nach demgegenüber keinerlei Vorteile.

LISTEN

enumitem

Das Paket **enumitem** erweitert die rudimentären Funktionalitäten der L^AT_EX-Standardlisten `itemize`, `enumerate` sowie `description` und ermöglicht die individuelle Anpassung dieser durch die Bereitstellung vieler optionaler Parameter nach dem Schlüssel-Wert-Prinzip. Eine von mir sehr häufig genutzte Funktion ist beispielsweise die Entfernung des zusätzlichen Abstand zwischen den einzelnen Einträgen einer Liste mit `\setlist{noitemsep}`.

TABELLEN

Für den Tabellensatz in L^AT_EX werden von Haus aus die Umgebungen `tabbing` und `tabular` beziehungsweise `tabular*` bereitgestellt, welche in ihrer Funktionalität meist für einen qualitativ hochwertigen Tabellensatz nicht ausreichen. Es werden deshalb Pakete vorgestellt, die zusätzlich verwendet werden können.

array

Dieses Paket ermöglicht mit dem Befehl `\newcolumntype` das Erstellen eigener Spaltentypen sowie die erweiterte Definition von Tabellenspalten (`>{...}<Spaltentyp><{...}<`), wobei mithilfe sogenannter „Hooks“ vor und nach Einträgen innerhalb einer Spalte gezielt Anweisungen gesetzt werden können. Außerdem kann die Höhe der Zeilen einer Tabelle mit `\extrarowheight` angepasst werden.

multirow

Es wird der Befehl `\multirow` definiert, der – ähnlich zum Makro `\multicolumn` – das Zusammenfassen von mehreren Zeilen in einer Spalte ermöglicht.

widetable

Mit der Standard- \LaTeX -Umgebung `tabular*` kann eine Tabelle mit einer definierten Breite gesetzt werden. Dieses Paket stellt die Umgebung `widetable` zur Verfügung, die als Alternative genutzt werden kann und eine symmetrische Tabelle erzeugt.

booktabs

Für einen guten Tabellensatz mit \LaTeX gibt es bereits zahlreiche [Tipps](#)⁶⁷ im Internet zu finden. Zwei Regeln sollten dabei definitiv beachtet werden:

- I. keine vertikalen Linien
- II. keine doppelten Linien

Das Paket `booktabs` (deutsche Dokumentation `booktabs-de`) ist für den Satz von hochwertigen Tabellen eine große Hilfe und stellt die Befehle `\toprule`, `\midrule` sowie `\cmidrule` und `\bottomrule` für unterschiedliche horizontale Linien bereit.

tabularborder

Bei Tabellen wird zwischen Spalten automatisch ein horizontaler Abstand (`\tabcolsep`) gesetzt – besser gesagt jeweils vor und nach einer Spalte. Dies geschieht auch *vor* der ersten und *nach* der letzten Spalte. Dieser zusätzliche Platz an den äußeren Rändern kann störend wirken, insbesondere wenn die Tabelle über die komplette Textbreite gesetzt wird. Mit dem Paket `tabularborder` kann dieser Platz automatisch entfernt werden.

Dies funktioniert allerdings nur mit der `tabular`-Umgebung. Die Tabellen aus den Paketen `tabularx`, `tabulary` und `tabu` werden nicht unterstützt. Wie dieser Abstand bei diesen manuell entfernt werden kann, ist unter [Unterkapitel 7.9](#) zu finden.

tabularx

Auch mit diesem Paket kann die Gesamtbreite einer Tabelle spezifiziert werden. Dafür wird der Spaltentyp `X` definiert, welcher als Argument der `tabularx`-Umgebung beliebig häufig angegeben werden kann (`\begin{tabularx}{<Breite>}{<Spalten>}`). Die `X`-Spalten ähneln denen vom Typ `p{<Breite>}`, wobei die Breite dieser aus der gewünschten Tabellengesamtbreite und dem benötigten Platz der gegebenenfalls vorhandenen Standardspalten automatisch berechnet wird.

tabulary

Dies ist ein weiteres Paket zur automatischen Berechnung von Spaltenbreiten. Der zur Verfügung stehende Platz – gewünschte Gesamtbreite abzüglich der notwendigen Breite für die Standardspalten – wird jedoch nicht wie bei der Umgebung `tabularx` auf alle Spalten gleichmäßig verteilt sondern in der `tabulary`-Umgebung für die Spaltentypen `L`, `C`, `R` und `J` anhand ihres Zellinhaltes gewichtet vergeben. (`\begin{tabulary}{<Breite>}{<Spalten>}`).

longtable, ltablex

Sollen mehrseitige Tabellen mit Seitenumbruch erstellt werden, ist dieses Paket das Mittel der ersten Wahl. Für die Kombination mehrseitiger Tabellen mit einer `tabularx`-Umgebung können die Pakete `ltablex` oder besser noch `ltxtable` verwendet werden.

ltxtable

Wie bereits erwähnt sollte dieses Paket für mehrseitige Tabellen, die mit der Umgebung `tabularx` erstellt wurden, verwendet werden. Alternativ dazu kann man auch `tabu` nutzen.

⁶⁷<http://userpage.fu-berlin.de/latex/Materialien/tabsatz.pdf>

Dies ist ein relativ neues Paket, welches versucht, viele der zuvor genannten Funktionalitäten zu implementieren und weitere bereitzustellen. Dafür werden die Umgebungen `tabu` und `longtabu` definiert. Es kann alternativ zu `tabularx` verwendet werden und ist insbesondere als Ersatz für das Paket `ltxtable` empfehlenswert.

Achtung!

*Leider wären für das Paket in der aktuellen Version v2.8 seit geraumer Zeit ein paar kleinere Bugfixes notwendig. Außerdem wird sich die **Benutzerschnittstelle in einer zukünftigen Version** ⁶⁸ sehr stark ändern, weshalb es momentan mit Vorsicht zu genießen ist. Zumindest sollte sich der Anwender bewusst sein, dass er mit dieser Version gesetzte Dokumente gegebenenfalls später anpassen muss.*

GLEITOBJEKTE

Es werden Pakete für die Beeinflussung von Aussehen, Beschriftung und Positionierung von Gleitobjekten vorgestellt. Unter [Unterkapitel 7.20](#) sind außerdem Hinweise zur manuellen Manipulation der Gleitobjektplatzierung zu finden.

placeins

Mit diesem Paket kann die Ausgabe von Gleitobjekten vor Kapiteln und wahlweise Unterkapiteln erzwungen werden.

flafter

Dieses Paket erlaubt die frühestmögliche Platzierung von Gleitobjekten im ausgegeben Dokument erst an der Stelle ihres Auftretens im Quelltext. Diese werden dementsprechend nie vor ihrer Definition am Anfang der Seite erscheinen.

caption

Die KOMA-Script-Klassen bietet bereits einige Möglichkeiten zum Setzen der Beschriftungen für Gleitobjekte. Dieses Paket ist daher meist nur in gewissen Ausnahmefällen für spezielle Anweisungen notwendig, allerdings auch bei der Verwendung unbedenklich.

subcaption

Dieses Paket kann zum einfachen Setzen von Unterabbildungen oder -tabellen mit den entsprechenden Beschriftungen genutzt werden. Das dazu alternative Paket `subfig` sollte vermieden werden, da es nicht mehr gepflegt wird und es mit diesem im Zusammenspiel mit anderen Paketen des Öfteren zu Problemen kommt. Sollte der Funktionsumfang von `subcaption` nicht ausreichen, kann anstelle dessen das Paket `floatrow` verwendet werden, welches ähnliche Funktionalitäten wie `subfig` bereitstellt.

floatrow

Mit diesem Paket können global wirksame Einstellungen und Formatierungen für *alle* Gleitobjekte eines Dokumentes vorgenommen werden. So kann unter anderem die verwendete Schrift (`\floatsetup{font=...}`) innerhalb der Umgebungen `float` und `table` eingestellt werden. Das typografisch richtige Setzen der Beschriftungen von Abbildungen als Unterschriften (`\floatsetup[figure]{capposition=bottom}`) sowie Tabellen als Überschriften (`\floatsetup[table]{capposition=top}`) kann automatisch erzwungen werden – unabhängig von der Position des Befehls zur Beschriftung `\caption` innerhalb der Gleitobjektumgebung. Wird das Verhalten so wie empfohlen mit dem `floatrow`-Paket eingestellt, sollte für eine richtige Platzierung der Tabellenüberschriften außerdem die KOMA-Script-Option `captions=tableheading` genutzt werden.

⁶⁸<https://groups.google.com/d/topic/comp.text.tex/xRGJTC74uCI>

GRAFIKEN UND ABBILDUNGEN

Grafiken für wissenschaftliche Arbeiten sollten als Vektorgrafiken erstellt werden, um die Skalierbarkeit und hohe Druckqualität zu gewährleisten. Bestenfalls folgen diese auch dem Stil der dazugehörigen Arbeit.⁶⁹ Für das Erstellen eigener Vektorgrafiken, welche die \LaTeX -Schriften und das Layout des Hauptdokumentes nutzen, gibt es zwei mögliche Ansätze. Entweder man „programmiert“ die Grafiken ähnlich wie das Dokument selber oder man nutzt Zeichenprogramme, die wiederum die Ausgabe oder das Weiterreichen von Text an \LaTeX unterstützen. Für das Programmieren von Grafiken sollen hier die wichtigsten Pakete vorgestellt werden. Wie diese zu verwenden sind, ist den dazugehörigen Paketdokumentationen zu entnehmen. Außerdem wird im Tutorial [treatise.pdf](#) für beide Pakete jeweils ein Beispiel gegeben.

tikz

Dies ist ein sehr mächtiges Paket für das Programmieren von Vektorgrafiken und sehr häufig – insbesondere bei Einsteigern – die erste Wahl bei der Verwendung von pdf \LaTeX .

pstricks

Das Paket [pstricks](#) stellt die zweite Variante zum Programmieren von Grafiken dar. Mit diesem Paket hat man *noch* mehr Möglichkeiten bei der Erstellung eigener Grafiken, da man mit [pstricks](#) auf PostScript zugreifen kann und einige der bereitgestellten Befehle davon rege Gebrauch machen. Der daraus resultierende Nachteil ist, dass mit [pstricks](#) die direkte Verwendung von pdf \LaTeX nicht möglich ist.

Die Grafiken aus den `pspicture`-Umgebungen müssen deshalb erst über den Pfad `latex → dvips → ps2pdf` in PDF-Dateien gewandelt werden. Diese lassen sich von pdf \LaTeX anschließend als Abbildungen einbinden. Um dieses Vorgehen zu ermöglichen, können folgende Pakete genutzt werden:

pst-pdf

Dieses Paket stellt Methoden für den Export von PostScript-Grafiken in PDF-Dateien bereit. Die einzelnen Aufrufe zur Kompilierung von DVI über PostScript zu PDF müssen durch den Anwender manuell beziehungsweise über die Ausgaberoutinen des verwendeten Editors durchgeführt werden.

auto-pst-pdf, pdftricks2

Das Paket automatisiert die Erzeugung der [pstricks](#)-Grafiken mit dem Paket [pst-pdf](#). Dafür muss pdf \LaTeX per Option mit Schreibrechten ausgeführt werden. Dazu ist der Aufruf von `pdflatex` mit der Option `--shell-escape` beziehungsweise für Nutzer von **MiKTeX** mit `--enable-write18` notwendig. Bitte beachten Sie die Hinweise in [Unterkapitel 7.24](#). Eine Alternative dazu ist das Paket [pdftricks2](#).

Um bei der Erstellung von Grafiken mit [pstricks](#) oder [tikz](#) nicht bei jeder Änderung das komplette Dokument kompilieren zu müssen, können diese in separate Dateien ausgelagert werden. Hierfür sind die beiden Pakete [standalone](#) oder [subfiles](#) sehr nützlich.

Für das Zeichnen einer Grafik mit einem Bildbearbeitungsprogramm, welches die Weiterverarbeitung durch \LaTeX erlaubt, möchte ich auf die freien Programme **LaTeXDraw** und **Inkscape** verweisen. Insbesondere das zuletzt genannte Programm ist sehr empfehlenswert. Für die erstellten Grafiken kann man den Export für die Einbindung in \LaTeX manuell durchführen. In [Unterkapitel 7.21](#) wird vorgestellt, wie sich dies automatisieren lässt.

⁶⁹Für qualitativ hochwertige Dokumente sollten übernommene Grafiken nicht direkt kopiert oder gescannt sondern im gewünschten Zielformat neu erstellt und mit der Referenz auf die Quelle ins Dokument eingebunden werden.

AUFTEILUNG DES HAUPTDOKUMENTES IN UNTERDATEIEN

Um während des Entwurfes eines Dokumentes die Zeitdauer für das Kompilieren zu verkürzen, kann dieses in Unterdokumente gegliedert werden. Dadurch wird es möglich, nur den momentan bearbeiteten Dokumentteil – respektive die aktuelle **tikz**- oder **pstricks**-Grafik – zu kompilieren. Die meiner Meinung nach besten Pakete für dieses Unterfangen werden folgend vorgestellt.

v2.02

standalone

Dieses Paket ist für das Erstellen eigenständiger (Unter-)Dokumente gedacht, welche später in ein Hauptdokument eingebunden werden können. Jedes dieser Teildokumente benötigt eine eigene Präambel. Optional lassen sich die Präambeln der Unterdokumente automatisch in ein Hauptdokument einbinden.

v2.02

subfiles

Dieses Paket wählt einen etwas anderen Ansatz als **standalone**. Es ist von Anfang an dafür gedacht, ein dediziertes Hauptdokument zu verwenden. Die darin mit `\subfiles` eingebundenen Unterdateien nutzen bei der autarken Kompilierung dessen Präambel.

Unabhängig davon, ob Sie eines der beiden Pakete nutzen oder alles in einem Dokument belassen, ist es ratsam, eigens definierte Befehle, Umgebungen und ähnliches in ein separates Paket auszulagern. Dafür müssen Sie lediglich ein leeres \LaTeX -Dokument erzeugen und es unter **mypreamble.sty** oder einem anderen Namen im gleichen Ordner wie das Hauptdokument speichern. Dann können Sie in dieser Datei ihre Deklarationen ausführen und diese mit `\usepackage{mypreamble}` in das Dokument einbinden. Dies hat den Vorteil, dass das Hauptdokument zum einen übersichtlich bleibt und Sie zum anderen Ihre persönliche Präambel generisch wachsen lassen und für andere Dokumente wiederverwenden können.

DIE KLEINEN UND GROSSEN HELFER...

Hier taucht alles auf, was nicht in die vorherigen Kategorien eingeordnet werden konnte.

bookmark

Dieses Paket verbessert und erweitert die von **hyperref** angebotenen Möglichkeiten zur Erstellung von Lesezeichen – auch Outline-Einträge – im PDF-Dokument. Beispielsweise können Schriftfarbe- und -stil geändert werden.

calc

Normalerweise können Berechnungen nur mit Low-Level- \TeX -Primitiven im Dokument durchgeführt werden. Dieses Paket stellt eine einfachere Syntax für Rechenoperationen der vier Grundrechenarten zur Verfügung. Zusätzlich werden neue Befehle zur Bestimmung der Höhe und Breite bestimmter Textauszüge definiert.

chngcntr

Das Paket erlaubt die Manipulation aller möglichen, bereits definierten \LaTeX -Zähler. Es können Zähler so umdefiniert werden, dass sie bei der Änderung eines anderen Zählers automatisch zurückgesetzt werden oder eben nicht. Ein kleines Beispiel dazu ist in [Unterkapitel 7.11](#) zu finden.

varioref

Mit diesem Paket lassen sich sehr gute Verweise auf bestimmte Seiten erzeugen. Insbesondere, wenn der Querverweis auf die aktuelle, die vorhergehende oder nachfolgende sowie im zweiseitigen Satz auf die gegenüberliegende Seite erfolgt, werden passende Textbausteine für diesen verwendet.

cleveref

Dieses Paket vereint die Vorzüge von **varioref** mit der automatischen Benennung der referenzierten Objekte mit dem Befehl `\autoref` aus dem Paket **hyperref**.

marginnote

Randnotizen, welche mit `\marginpar` erzeugt werden, sind spezielle Gleitobjekte in \LaTeX . Dies kann dazu führen, dass eine Notiz am Blattrand nicht direkt da gesetzt wird, wo diese intendiert war. Dieses Paket stellt den Befehl `\marginnote` für nicht-gleitende Randnotizen zur Verfügung. Alternativ dazu kann man auch **mparhack** verwenden.

todonotes

Mit **todonotes** können noch offene Aufgaben in unterschiedlicher Formatierung am Blattrand oder im direkt Fließtext ausgegeben werden. Aus allen Anmerkungen lässt sich eine Liste aller offenen Punkte erzeugen.

listings

Dieses Paket eignet sich hervorragend zur Quelltextdokumentation in \LaTeX . Es bietet die Möglichkeit, externe Quelldateien einzulesen und darzustellen sowie die Syntax in Abhängigkeit der verwendeten Programmiersprache hervorzuheben. Zusätzlich lässt sich ein Verzeichnis mit allen eingebundenen sowie direkt im Dokument angegebenen Quelltextauszügen erstellen. Wird **listings** in Dokumenten mit UTF-8-Kodierung verwendet, sollte direkt nach dem Laden des Paketes in der Präambel Folgendes hinzugefügt werden:

```
\lstset{%
  inputencoding=utf8,extendedchars=true,
  literate=%
  {ä}{\{"a}}1 {ö}{\{"o}}1 {ü}{\{"u}}1
  {Ä}{\{"A}}1 {Ö}{\{"O}}1 {Ü}{\{"U}}1
  {~}{\{\textasciitilde}}1 {\ß}{\{\ss}}1
}
```

xparse

Dieses mächtige Paket entstammt dem \LaTeX 3-Projekt und bietet für die Erstellung eigener Befehle und Umgebungen einen alternativen Ansatz zu den bekannten \LaTeX -Deklarationsbefehlen `\newcommand` und `\newenvironment` sowie deren Derivaten. Mit **xparse** wird es möglich, obligatorische und optionale Argumente an beliebigen Stellen innerhalb des Befehlskonstruktes zu definieren. Auch die Verwendung anderer Zeichen als eckige Klammern für die Spezifizierung eines optionalen Argumentes ist möglich.

v2.02

xkeyval, keyval, kvoptions, pgfkeys

Das KOMA-Script-Bundle lädt das Paket **keyval**, um Optionen mit einer Schlüssel-Wert-Syntax deklarieren zu können. Zusätzlich wird von TUD-KOMA-Script das Paket **kvsetkeys** geladen, um auf nicht definierte Schlüssel reagieren zu können. Die Schlüssel-Wert-Syntax kann auch für eigens definierte Makros genutzt werden, um sich das exzessive Verwenden von optionalen Argumenten zu ersparen. Damit wäre folgende Definition möglich: `\newcommand\Befehl[<Schlüssel-Wert-Liste>]{<Argument>}`.

Das Paket **xkeyval** erweitert insbesondere die Möglichkeiten zur Deklaration unterschiedlicher Typen von Schlüsseln. Sollten die bereits durch TUD-KOMA-Script geladenen Pakete **keyval** und **kvsetkeys** in ihrer Funktionalität nicht ausreichen, kann dieses Paket verwendet werden. Für die Entwicklung eigener Pakete, deren Optionen das Schlüssel-Wert-Format unterstützen, kann das Paket **scrbase** genutzt werden. Soll aus einem Grund auf KOMA-Script gänzlich verzichtet werden, sind die beiden Pakete **kvoptions** oder **pgfkeys** eine Alternative.

afterpage

Der Befehl `\afterpage{<...>}` kann genutzt werden, um den Inhalt aus dessen Argument direkt nach der Ausgabe der aktuellen Seite auszuführen.

pagecolor

Mit dem Paket kann die Hintergrundfarbe der Seiten im Dokument geändert werden.

pdfpages

Das Paket ermöglicht die Einbindung von einzelnen oder mehreren PDF-Dateien.

v2.02

mwe

Mit diesem Paket lassen sich sehr einfach Minimalbeispiele erzeugen, die sowohl Blindtexte respektive Abbildungen enthalten sollen.

filemod

Wird entweder pdf \LaTeX oder Lua \LaTeX als Prozessor eingesetzt, können mit diesem Paket das Änderungsdatum zweier Dateien miteinander verglichen und in Abhängigkeit davon definierbare Aktionen ausgeführt werden.

coseoul

Mit diesem Paket kann man die Struktur der Gliederung relativ angeben. Es wird keine absolute Gliederungsebene (`\chapter`, `\section`) angegeben sondern die Relation zwischen vorheriger und aktueller Ebene (`\levelup`, `\levelstay`, `\leveldown`).

dprogress

Das Paket schreibt bei der Kompilierung des Dokumentes die Gliederung in die Logdatei. Dies kann im Fehlerfall beim Auffinden des Problems im Dokument helfen. Allerdings werden dafür die Gliederungsebenen so umdefiniert, dass diese keine optionalen Argumente mehr unterstützen, was jedoch für die TUD-KOMA-Script-Klassen von essentieller Bedeutung ist. Zum Debuggen kann es trotzdem sporadisch eingesetzt werden.

BUGFIXES

scrhack

Das Paket behebt Kompatibilitätsprobleme der KOMA-Script-Klassen mit den Paketen [hyperref](#), [float](#), [floatrow](#) und [listings](#). Es ist durchaus empfehlenswert, jedoch sollte man unbedingt die Dokumentation beachten.

mparhack

Zur Behebung falsch gesetzter Randnotizen wird ein Bugfix für `\marginpar` bereitgestellt. Alternativ dazu kann man auch [marginnote](#) verwenden.

KAPITEL 7

PRAKTISCHE TIPPS & TRICKS

7.1 L^AT_EX-EDITOREN

Hier werden die gängigsten Editoren zum Erzeugen von L^AT_EX-Dateien genannt. Ich persönlich bin mittlerweile sehr überzeugter Nutzer von *TeXstudio*, da dieser viele Unterstützungs- und Assistenzfunktionen bietet. Neben diesen gibt es noch weitere, gut nutzbare L^AT_EX-Editoren. Egal, für welchen Editor man sich letztendlich entscheidet, sollte dieser auf jeden Fall eine Unicode-Unterstützung (UTF-8) enthalten:

- *TeXmaker*
- *Kile*
- *TeXworks*
- *TeXlipse* – Plug-in für *Eclipse*
- *TeXnicCenter*
- *WinEdt*
- *LEd* – früher L^AT_EX Editor
- *LyX* – grafisches Front-End für L^AT_EX

Für *TeXstudio* wird im [GitHub-Repository tudscr](#) das Archiv [tudscr4texstudio.zip](#)⁷⁰ bereitgestellt, welches Dateien zur Erweiterung der automatischen Befehlsvervollständigung für TUD-KOMA-Script enthält. Diese müssen unter Windows in %APPDATA%\texstudio beziehungsweise unter unixoiden Betriebssystemen in .config/texstudio eingefügt werden.

Möchten Sie das grafische L^AT_EX-Frontend *LyX* für das Erstellen eines Dokumentes mit TUD-KOMA-Script nutzen, so werden dafür spezielle Layout-Dateien benötigt, um die Klassendateien verwenden zu können. Diese sind zusammen mit einem *LyX*-Dokument als Archiv [tudscr4lyx.zip](#)⁷¹ im [GitHub-Repository tudscr](#) verfügbar. Die Layout-Dateien müssen dafür im *LyX*-Installationspfad in den passenden Unterordner kopiert werden. Dieser ist bei Windows %PROGRAMFILES(X86)%\LyX 2.1\Resources\layouts beziehungsweise bei unixoiden Betriebssystemen /usr/share/lyx/layouts. Anschließend muss *LyX* über den Menüpunkt *Werkzeuge* neu konfiguriert werden.

7.2 LITERATURVERWALTUNG IN L^AT_EX

v2.02

Die simpelste Variante, eine L^AT_EX-Literaturdatenbank zu verwalten, ist dies mit dem Editor manuell zu erledigen. Wesentlich komfortabler ist es jedoch, die Referenzverwaltung mit einer darauf spezialisierten Anwendung zu bewerkstelligen. Dafür gibt es zwei sehr gute Programme:

- *Citavi*
- *JabRef*

Das Programm *Citavi* ermöglicht den Import von bibliografischen Informationen aus dem Internet. Allerdings sind diese teilweise unvollständig oder mangelhaft. Mit *JabRef* hingegen muss die Literaturdatenbank manuell erstellt werden. Allerdings lassen sich einzelne Einträge aus .bib-Dateien sehr importieren. Beide Anwendungen unterstützen den Export beziehungsweise die Erstellung von Datenbanken im Stil von [biblatex](#). Für *JabRef* muss diese durch den Anwender explizit aktiviert werden.⁷² Zur Verwendung der beiden Programme in Verbindung mit [biblatex](#) und *biber* gibt es ein gutes Tutorial unter diesem [Link](#).

⁷⁰<https://github.com/tud-cd/tudscr/releases/download/TeXstudio/tudscr4texstudio.zip>

⁷¹<https://github.com/tud-cd/tudscr/releases/download/LyX/tudscr4lyx.zip>

⁷²Optionen/Einstellungen/Erweitert/BibLaTeX-Modus

7.3 WORTTRENNUNGEN IN DEUTSCHSPRACHIGEN TEXTEN

v2.02

Die möglichen Trennstellen von Wörtern werden von $\text{\LaTeX} 2_{\epsilon}$ mithilfe eines Algorithmus berechnet. Dieser ist jedoch in seiner ursprünglichen Form für die englische Sprache konzipiert worden. Für deutschsprachige Texte wird die Worttrennung – insbesondere bei zusammengesetzten Wörtern – mit dem Paket **hyphsubst** entscheidend verbessert. Dafür wird ein um vielerlei Trennungsmuster ergänztes Wörterbuch aus dem Paket **dehyph-exptl** genutzt.

Das Paket **hyphsubst** muss bereits *vor* der Dokumentklasse selbst geladen werden. Außerdem wird das Paket **babel** benötigt. Damit auch Wörter mit Umlauten richtig getrennt werden, ist zusätzlich die Verwendung des Paketes **fontenc** mit der T1-Schriftkodierung erforderlich. Der Beginn einer Dokumentpräambel könnte folgendermaßen aussehen:

```
\RequirePackage[ngerman=ngerman-x-latest]{hyphsubst}
\documentclass[ngerman,<Klassenoptionen>]{<Dokumentklasse>}
\usepackage{selinput}\SelectInputMappings{adieresis={ä},germandbls={ß}}
\usepackage[T1]{fontenc}
\usepackage{babel}
...
```

Eine Anmerkung noch zur Trennung von Wörtern mit Bindestrichen. Normalerweise sind die beiden von $\text{\LaTeX} 2_{\epsilon}$ verwendeten Zeichen für Bindestrich und Trennstrich identisch. Leider wird der Trennungsalgorithmus von $\text{\LaTeX} 2_{\epsilon}$ bei Wörtern, welche bereits einen Bindestrich enthalten, außer Kraft gesetzt. In der Folge werden – in der deutschen Sprache durchaus öfter anzutreffende – Wortungetüme wie die „Donaudampfschiffahrts-Gesellschafterversammlung“ normalerweise nur direkt nach dem angegebenen Bindestrich getrennt.

Allerdings gibt es die Möglichkeit, das genutzte Zeichen für den Trennstrich zu ändern. Dafür ist das Laden der T1-Schriftkodierung mit dem Paket **fontenc** zwingend erforderlich. Wenn von der verwendeten Schrift nichts anderes eingestellt ist, liegen sowohl Binde- als auch Trennstrich auf Position 45 der Zeichentabelle. In der T1-Schriftkodierung befindet sich auf der Position 127 glücklicherweise für gewöhnlich das gleiche Zeichen noch einmal. Dies ist jedoch von der verwendeten Schrift abhängig. Wird der Ausdruck `\defaultshyphenchar=127` in der Dokumentpräambel verwendet, kann dieses Zeichen für den Trennstrich genutzt werden. Bei den Schriften des Corporate Designs der Technischen Universität Dresden ist dies bereits automatisch eingestellt.

Sollte trotz aller Maßnahmen dennoch einmal ein bestimmtes Wort falsch getrennt werden, so kann die Worttrennung dieses Wortes manuell und global geändert werden. Dies wird mit `\hyphenation{Sil-ben-tren-nung}` gemacht. Es ist zu beachten, dass dies für alle Flexionsformen des Wortes erfolgen sollte. Für eine lokale/temporäre Worttrennung kann mit Befehlen aus dem Paket **babel** gearbeitet werden. Diese sind:

Beschreibung	Befehl
ausschließliche Trennstellen	<code>\-</code>
zusätzliche Trennstellen	<code>"-</code>
Umbruch ohne Trennstrich	<code>" "</code>
Bindestrich ohne Umbruch	<code>"~</code>
Bindestrich, der weitere Trennstellen erlaubt	<code>"=</code>

7.4 LOKALE ÄNDERUNGEN VON BEFEHLEN UND EINSTELLUNGEN

v2.02

Ein zentraler Bestandteil von \LaTeX ist die Verwendung von Gruppen oder Gruppierungen. Innerhalb dieser bleiben alle vorgenommenen Änderungen an Befehlen, Umgebungen oder Einstellungen lokal. Dies kann sehr nützlich sein, wenn beispielsweise das Verhalten eines bestimmten Makros einmalig oder innerhalb von selbst definierten Befehlen oder Umgebungen geändert werden, im Normalfall jedoch die ursprüngliche Funktionalität behalten soll.

Beispiel: Der Befehl `\emph` wird von \LaTeX für Hervorhebungen im Text bereitgestellt und führt normalerweise zu einer kursiven oder – falls kein Schriftschnitt mit echten Kursiven vorhanden ist – kursivierten Auszeichnung. Soll nun in einem bestimmten Abschnitt die Auszeichnung mit fetter Schrift erfolgen, kann der Befehl `\emph` innerhalb einer Gruppierung geändert und verändert werden. Wird diese beendet, verhält sich der Befehl wie gewohnt.

In diesem Text wird ein bestimmtes `\emph{Wort}` hervorgehoben.

```
\begingroup
\renewcommand*\emph[1]{\textbf{#1}}%
In diesem Text wird ein bestimmtes \emph{Wort} hervorgehoben.
\endgroup
```

In diesem Text wird ein bestimmtes `\emph{Wort}` hervorgehoben.

Eine Gruppierung kann entweder mit `\begingroup` und `\endgroup` oder einfach mit einem geschweiften Klammerpaar `{...}` definiert werden.

7.5 BEZEICHNUNG DER GLIEDERUNGSEBENEN DURCH HYPERREF

v2.02

Das Paket [hyperref](#) stellt für Querverweise unter anderem den Befehl `\autoref{<label>}` zur Verfügung. Mit diesem wird – im Gegensatz zur Verwendung von `\ref` – bei einer Referenz nicht nur die Nummerierung selber sondern auch das entsprechende Element wie Kapitel oder Abbildung vorangestellt. Bei der Benennung des referenzierten Elementes wird sequentiell geprüft, ob das Makro `<Element>autorefname` oder `<Element>name` existiert. Soll die Bezeichnung eines Elementes geändert werden, muss man den entsprechende Bezeichner anpassen.

Beispiel: Bezeichnungen von Gliederungsebenen können folgendermaßen verändert werden.

```
\renewcaptionname{ngerman}{\sectionautorefname}{Unterkapitel}
\renewcaptionname{ngerman}{\subsectionautorefname}{Abschnitt}
\renewcaptionname{ngerman}{\subsubsectionautorefname}{Unterabschnitt}
```

7.6 URL-UMBRÜCHE IM LITERATURVERZEICHNIS MIT BIBLATEX

v2.02

Wird das Paket [biblatex](#) verwendet, kann es unter Umständen dazu kommen, dass eine URL nicht vernünftig umbrochen werden. Ist dies der Fall, können die Zähler `biburlnumpenalty`, `biburlucpenalty` und `biburllcpenalty` erhöht werden. Die möglichen Werte liegen zwischen 0 und 10000, wobei es bei höheren Werten der Zähler zu mehr URL-Umb Brüchen an Ziffern (`biburlnumpenalty`), Groß- (`biburlucpenalty`) und Kleinbuchstaben (`biburllcpenalty`) kommt. Genauer hierzu ist der Dokumentation des [biblatex](#)-Paketes zu entnehmen.

7.7 ZEILENABSTÄNDE IN ÜBERSCHRIFTEN

Mit dem Paket `setspace` kann der Zeilenabstand beziehungsweise der Durchschuss innerhalb des Dokumentes geändert werden. Sollte dieser erhöht worden sein, können die Abstände bei mehrzeiligen Überschriften als zu groß erscheinen. Um dies zu korrigieren kann mit dem Befehl `\addtokomafont{disposition}{\setstretch{1}}` der Zeilenabstand aller Überschriften auf einzeilig zurückgeschaltet werden. Soll dies nur für eine bestimmte Gliederungsebene erfolgen, so ist der Parameter `disposition` durch das dazugehörige Schriftelement zu ersetzen.

7.8 WARNUNG WEGEN ZU GERINGER HÖHE DER KOPF-/FUSSZEILE

Wird das Paket `setspace` verwendet, kann es passieren, dass nach der Änderung des Zeilenabstandes *innerhalb* des Dokumentes eine oder beide der folgenden Warnungen erscheinen:

```
scrlayer-scrpage Warning: \headheight to low.  
scrlayer-scrpage Warning: \footheight to low.
```

Dies liegt an dem durch den vergrößerten Zeilenabstand erhöhten Bedarf für die Kopf- und Fußzeile, die Höhen können in diesem Fall direkt mit der Verwendung von `\recalctypearea` angepasst werden. Allerdings ändert das den Satzspiegel im Dokument, was eine andere und durchaus berechnete Warnung von `typearea` zur Folge hat. Falls die Änderung des Durchschusses wirklich nötig ist, sollte dies in der Präambel des Dokumentes einmalig passieren. Dann entfallen auch die Warnungen.

7.9 EINRÜCKUNG VON TABELLENSPALTEN VERHINDERN

Normalerweise wird in einer Tabelle vor *und* nach jeder Spalte durch $\text{\LaTeX 2}_{\epsilon}$ etwas horizontaler Raum mit `\hskip\tabcolsep` eingefügt.⁷³ Dies geschieht auch *vor* der ersten und *nach* der letzten Spalte. Diese optische Einrückung an den äußeren Rändern kann unter Umständen stören, insbesondere bei Tabellen, die willentlich – beispielsweise mit den Paketen `tabularx`, `tabulary` oder auch `tabu` – über die komplette Seitenbreite aufgespannt werden.

Das Paket `tabularborder` versucht, dieses Problem automatisiert zu beheben, ist jedoch nicht zu allen $\text{\LaTeX 2}_{\epsilon}$ -Paketen für den Tabellensatz kompatibel, unter anderem auch nicht zu den drei zuvor genannten. Allerdings lässt sich dieses Problem manuell durch den Anwender lösen.

Bei der Deklaration einer Tabelle kann mit `@{...}` vor und nach dem Spaltentyp angegeben werden, was anstelle von `\tabcolsep` vor beziehungsweise nach der eigentlichen Spalte eingeführt werden soll. Dies kann für das Entfernen der Einrückungen genutzt werden, indem `@{}` an den entsprechenden Stellen bei der Angabe der Spaltentypen vor der ersten und nach der letzten Tabellenspalte verwendet wird.

Beispiel: Eine Tabelle mit zwei Spalten, wobei bei einer die Breite automatisch berechnet wird, soll über die komplette Textbreite gesetzt werden. Dabei soll der Rand vor der ersten und nach der letzten entfernt werden.

```
\begin{tabularx}{\textwidth}{@{}lX@{}}  
... & ... \tabularnewline  
...  
\end{tabularx}
```

⁷³Der Abstand zweier Spalten beträgt folglich `2\tabcolsep`.

7.10 UNTERDRÜCKUNG DES EINZUGES EINES ABSATZES

Werden zur Auszeichnung von Absätzen im Dokument – wie es aus typografischer Sicht zumeist sinnvoll ist – Einzüge und keine vertikalen Abstände verwendet (KOMA-Script-Option `parskip=false`), kann es vorkommen, dass ein ganz bestimmter Absatz – beispielsweise der nach einer zuvor genutzten Umgebung folgende – ungewollt eingerückt ist. Dies kann sehr einfach behoben werden, indem direkt zu Beginn des Absatzes das Makro `\noindent` aufgerufen wird. Möchte man das Einrücken von Absätzen nach ganz bestimmten Umgebungen oder Befehlen automatisiert unterbinden, ist das Paket [noindentafter](#) zu empfehlen.

7.11 UNTERBINDEN DES ZURÜCKSETZENS VON FUSSNOTEN

Oft taucht die Frage auf, wie man auch über Kapitel fortlaufende Fußnoten erhalten kann. Dies ist sehr einfach mit dem Paket [chngecntr](#) möglich. Nach dem Laden des Paketes, kann das Rücksetzen des Zählers nach einem Kapitel mit `\counterwithout*{footnote}{chapter}` deaktiviert werden. Auch andere \LaTeX -Zähler – wie beispielsweise der bereits vorgestellte [symbolheadings](#) – lassen sich mit diesem Paket manipulieren.

7.12 SETZEN VON EINHEITEN MIT SIUNITX

Wenn [siunitx](#) in einem deutschsprachigen Dokument genutzt soll werden, muss zumindest mit `\sisetup{locale = DE}` die richtige Lokalisierung angegeben werden. Sollen auch die Zahlen richtig formatiert sein, müssen weitere Einstellungen vorgenommen werden. Die meiner Meinung nach besten sind die folgenden.

```
\sisetup{%
  locale = DE,%
  input-decimal-markers={,},input-ignore={.},%
  group-separator={\,},group-minimum-digits=3%
}
```

Das Komma kommt als Dezimaltrennzeichen zum Einsatz. Des Weiteren werden Punkte innerhalb der Zahlen ignoriert und eine Gruppierung von jeweils drei Ziffern vorgenommen. Alternativ zu diesem Paket kann übrigens auch [units](#) verwendet werden.

7.13 WARNUNG BEIM ERZEUGEN DES INHALTSVERZEICHNISSES

v2.02

Erstellt man ein Inhaltsverzeichnis für ein Dokument mit einer dreistelligen Seitenanzahl, so erhält man bei der Verwendung von `\tableofcontents` unter Umständen viele Warnungen mit der Meldung:

```
overfull \hbox
```

Das liegt daran, dass die Seitenzahlen in Verzeichnis in einer Box mit der Breite `\@pnumwidth` gesetzt werden. Der hierfür standardmäßig verwendete Wert von `1.55em` ist in diesem Fall zu klein und sollte vergrößert werden. Dabei ist auch der rechte Rand für mehrzeilige Einträge im Verzeichnis `\@pnumwidth` zu vergrößern, welcher mit `2.55em` voreingestellt ist. Die Anpassungen können folgendermaßen durchgeführt werden:

```

\makeatletter
\renewcommand*\@pnumwidth{1.7em}
\renewcommand*\@tocrmarg{2.7em}
\makeatother

```

Dabei sollte der eingesetzte Wert nicht zu groß ausfallen.

7.14 LEER- UND SATZZEICHEN NACH L^AT_EX-BEFEHLEN

Normalerweise „schluckt“ L^AT_EX die Leerzeichen nach einem Makro ohne Argumente. Dies ist jedoch nicht immer – genau genommen in den seltensten Fällen – erwünscht. Für dieses Handbuch ist beispielsweise der Befehl `\TUD` definiert worden, um „Technische Universität Dresden“ nicht ständig ausschreiben zu müssen. Um sich bei der Verwendung des Befehl innerhalb eines Satzes für den Erhalt eines folgenden Leerzeichens das Setzen der geschweiften Klammer nach dem Befehl zu sparen (`\TUD{}`), kann `\xspace` aus dem Paket `xspace` genutzt werden. Damit wird ein folgendes Leerzeichen erhalten. Der Befehl `\TUD` ist wie folgt definiert:

```
\newcommand*\TUD{Technische Universit\ "at Dresden\xspace}
```

Das Paket `xpunctuate` erweitert die Funktionalität nochmals. Damit können auch Abkürzungen so definiert werden, dass ein versehentlicher Punkt ignoriert wird:

```
\newcommand*\zB{z.\,B\xperiod}
```

7.15 DAS SETZEN VON AUSLASSUNGSPUNKTEN

v2.02

Auslassungspunkte werden mit L^AT_EX 2_ε mit den Befehlen `\dots` oder `\textellipsis` gesetzt. Für gewöhnlich folgt diesen *immer* ein Leerzeichen, was nicht in jedem Fall gewollt ist. Das Paket `ellipsis` schafft hier Abhilfe, wobei die Option `xspace` führt dazu, dass nach der Verwendung eines der beiden Befehle automatisch ein Leerzeichen gesetzt wird.

```
\usepackage[xspace]{ellipsis}
```

Im Ursprung ist es für das Setzen englischsprachiger Texte gedacht, wo zwischen Auslassungspunkten und Satzzeichen ein Leerzeichen gesetzt wird. Im Deutschen ist dies anders:

„Um eine Auslassung in einem Text zu kennzeichnen, werden drei Punkte gesetzt. Vor und nach den Auslassungspunkten wird jeweils ein Wortzwischenraum gesetzt, wenn sie für ein selbständiges Wort oder mehrere Wörter stehen. Bei Auslassung eines Wortteils werden sie unmittelbar an den Rest des Wortes angeschlossen. Am Satzende wird kein zusätzlicher Schlusspunkt gesetzt. Satzzeichen werden ohne Zwischenraum angeschlossen.“ [Duden, 23. Aufl.]

Um dieses Verhalten zu erreichen, sollte noch Folgendes in der Präambel eingefügt werden:

```

\let\ellipsis punctuation\relax
\newcommand*\qdots{[\dots{ }]\xspace}

```

Der Befehl `\qdots` wird definiert, um Auslassungspunkte in eckigen Klammern ([...]) setzen zu können, wie sie für das Kürzen von wörtlichen Zitaten häufig verwendet werden.

7.16 FINDEN VON UNBEKANNTEN L^AT_EX-SYMBOLEN

Für L^AT_EX stehen jede Menge Symbole zur Verfügung, die allerdings nicht immer einfach zu finden sind. In der Zusammenfassung [symbols-a4.pdf](http://mirrors.ctan.org/info/symbols/comprehensive/symbols-a4.pdf)⁷⁴ werden viele Symbole aus mehreren Paketen aufgeführt. Alternativ kann [Detexify](http://detexify.kirelabs.org/classify.html)⁷⁵ verwendet werden. Auf dieser Web-Seite wird das gesuchte Symbol einfach gezeichnet, die dazu ähnlichsten werden zurückgegeben.

7.17 ÄNDERUNG DES PAPIERFORMATES

Es kann vorkommen, dass man innerhalb eines Dokumentes kurzzeitig das Papierformat ändern möchte, um beispielsweise eine Konstruktionsskizze in der digitalen PDF-Datei einzubinden. Dabei ist es sowohl möglich, lediglich die Ausrichtung mit der KOMA-Script-Option `paper=landscape` in ein Querformat zu ändern, als auch die Größe des Papierformates selber.

Beispiel: Ein Dokument im A4-Format soll kurzzeitig auf ein A3-Querformat geändert werden. Das folgende Minimalbeispiel zeigt, wie das Papierformat mit den Mitteln von KOMA-Script geändert werden kann.

```
\documentclass[paper=a4,pagesize]{tudscrreprt}
\usepackage{selinput}
\SelectInputMappings{adieresis={ä},germandbls={ß}}
\usepackage[T1]{fontenc}
\usepackage[ngerman]{babel}
\usepackage{blindtext}

\begin{document}
\chapter{Überschrift Eins}
\Blindtext

\cleardoublepage
\storeareas\PotraitArea% speichert den aktuellen Satzspiegel
\KOMAOPTIONS{paper=A3,paper=landscape,DIV=current}
\chapter{Überschrift Zwei}
\Blindtext

\cleardoublepage
\PotraitArea% lädt den gespeicherten Satzspiegel
\chapter{Überschrift Drei}
\Blindtext
\end{document}
```

7.18 VERMEIDEN DES SKALIERENS EINER PDF-DATEI BEIM DRUCK

v2.04

Beim Erzeugen eines Druckauftrages einer PDF-Datei kann es unter Umständen dazu führen, dass diese durch den verwendeten PDF-Betrachter unnötigerweise vorher skaliert wird und dabei die Seitenränder vergrößert werden. Um dieses Verhalten für Dokumente, die mit pdfT_EX erzeugt werden, zu unterdrücken, gibt es zwei Möglichkeiten:

⁷⁴<http://mirrors.ctan.org/info/symbols/comprehensive/symbols-a4.pdf>

⁷⁵<http://detexify.kirelabs.org/classify.html>

1. Es kann `\hypersetup{pdfprintscaling=None}` verwendet werden, wenn im Dokument ohnehin das Paket `hyperref` geladen wird.
2. Der Low-Level-Befehl `\pdfcatalog{/ViewerPreferences<</PrintScaling/None>>}` hat das gleiche Verhalten und kann auch ohne das Paket `hyperref` genutzt werden.

Weitere Informationen dazu sind unter <http://www.komascript.de/node/1897> zu finden.

7.19 WARNUNG BEI DER SCHRIFTGRÖSSENWAHL

v2.04

Die im Dokument verwendete Schriftgröße kann bei den KOMA-Script-Klassen sehr einfach über die Option `fontsize` eingestellt werden, wobei diese immer als Klassenoption angegeben werden sollte. Bei relativ großen und kleinen Schriftgrößen kann dabei eine Warnung in der Gestalt

```
LaTeX Font Warning: Font shape '...' in size <xx> not available
```

auftreten. Dies daran, dass zum Zeitpunkt des Ladens einer Klasse immer nach den Computer-Modern-Standardschriften gesucht wird, unabhängig davon, ob im Nachhinein ein anderes Schriftpaket geladen wird. Diese sind de-facto nicht in alle Größen skalierbar. Um die Warnungen zu beseitigen, sollte das Paket `fix-cm` mit `\RequirePackage` vor der Dokumentklasse geladen werden:

```
\RequirePackage{fix-cm}
\documentclass[<Optionen>]{<Klasse>}
\usepackage{selinput}\SelectInputMappings{adieresis={ä},germandbls={ß}}
\usepackage[T1]{fontenc}
...
\begin{document}
...
\end{document}
```

Damit werden die Warnungen behoben.

7.20 PLATZIERUNG VON GLEITOBJEKTEN

Mit den beiden Paketen `flafter` sowie `placeins` gibt es die Möglichkeit, den für \LaTeX zur Verfügung stehenden Raum für die Platzierung von Gleitobjekten einzuschränken. Darüber hinaus kann diese auch durch die im Folgenden aufgezählten Befehle beeinflusst werden. Die Makros lassen sich mit `\renewcommand*{<Befehl>}{<Wert>}` sehr einfach ändern.

<code>\floatpagefraction</code>	<code>preset: 0.5</code>
<code>\dblfloatpagefraction</code>	<code>preset: 0.5</code>

Der Wert gibt die relative Größe eines Gleitobjektes bezogen auf die Texthöhe (`\textheight`) an, die mindestens erreicht sein muss, damit für dieses gegebenenfalls vor dem Beginn eines neuen Kapitels eine separate Seite erzeugt wird. Dabei wird einspaltiges (`\floatpagefraction`) und zweispaltiges (`\dblfloatpagefraction`) Layout unterschieden. Der Wert für beide Befehle sollte im Bereich von 0.5...0.8 liegen.

<code>\topfraction</code>	<code>preset: 0.7</code>
<code>\dbltopfraction</code>	<code>preset: 0.7</code>

Diese Werte geben den maximalen Seitenanteil für Gleitobjekte, die am oberen Seitenrand platziert werden, für einspaltiges und zweispaltiges Layout an. Er sollte im Bereich von 0.5...0.8 liegen und größer als `\floatpagefraction` beziehungsweise `\dblfloatpagefraction` sein.

<code>\bottomfraction</code>	<code>preset: 0.3</code>
------------------------------	--------------------------

Dies ist der maximale Seitenanteil für Gleitobjekte, die am unteren Seitenrand platziert werden. Er sollte zwischen 0.2 und 0.5 betragen.

<code>\textfraction</code>	<code>preset: 0.2</code>
----------------------------	--------------------------

Dies ist der Mindestanteil an Text, der auf einer Seite mit Gleitobjekten vorhanden sein muss, wenn diese nicht auf einer eigenen Seite ausgegeben werden. Er sollte im Bereich von 0.1...0.3 liegen.

<code>totalnumber</code> (Zähler)	<code>preset: 3</code>
<code>topnumber</code> (Zähler)	<code>preset: 2</code>
<code>dbltopnumber</code> (Zähler)	<code>preset: 2</code>
<code>bottomnumber</code> (Zähler)	<code>preset: 1</code>

Außerdem gibt es noch Zähler, welche die maximale Anzahl an Gleitobjekten pro Seite insgesamt (`totalnumber`), am oberen (`topnumber`) und am unteren Rand der Seite (`bottomnumber`) sowie im Zweispaltensatz beide Spalten überspannend (`dbltopnumber`) festlegen. Die Werte können mit `\setcounter{<Zähler>}{<Wert>}` geändert werden.

<code>\@fptop</code> (Länge)
<code>\@fpsep</code> (Länge)
<code>\@fpbot</code> (Länge)
<code>\@dblfpptop</code> (Länge)
<code>\@dblfpsep</code> (Länge)
<code>\@dblfpbot</code> (Länge)

Sind vor Beginn eines Kapitels noch Gleitobjekte verblieben, so werden diese durch L^AT_EX normalerweise auf einer separaten vertikal zentriert Seite ausgegeben. Dabei bestimmen diese Längen jeweils den Abstand vor dem ersten Gleitobjekt zum oberen Seitenrand (`\@fptop`, `\@dblfpptop`), zwischen den einzelnen Objekten (`\@fpsep`, `\@dblfpsep`) sowie zum unteren Seitenrand (`\@fpbot`, `\@dblfpbot`). Soll dies nicht geschehen, können die Längen durch den Anwender geändert werden.

Beispiel: Alle Gleitobjekte auf einer dafür speziell gesetzten Seite sollen direkt zu Beginn dieser ausgegeben werden. In der Dokumentpräambel kann man dafür schreiben:

```
\makeatletter
\setlength{\@fptop}{0pt}
\setlength{\@dblfpptop}{0pt} % twocolumn
\makeatother
```

7.21 AUTOMATISIERTES EINBINDEN VON INKSCAPE-GRAFIKEN

In [svg-inkscape](#)⁷⁶ wird das automatisierte Einbinden von *Inkscape*-Grafiken in ein \LaTeX -Dokument erläutert. Hier wird ein daraus abgeleiteter und verbesserter Ansatz vorgestellt. Nutzer von unixartigen Systemen können alternativ auch das Paket [svg](#) nutzen, welches den folgend erläuterten Befehl `\includesvg` definiert.

Die mit *Inkscape* erstellte Grafik soll automatisch kompiliert und eingebunden werden. Dies soll allerdings nicht bei jeder Kompilierung des Hauptdokumentes erfolgen, sondern lediglich, wenn die originale Bilddatei geändert beziehungsweise aktualisiert wurde. Hierfür wird [filemod](#) verwendet. Die automatisierte Übersetzung einer Grafik im SVG-Format in eine PDF-Datei und die daran anschließende Einbindung dieser in das Dokument ist mit der Definition von `\includesvg[<Breite>]{<Datei>}` in der Präambel des Dokumentes wie folgt möglich:

```
\usepackage{filemod}
\newcommand*{\includesvg}[2][\textwidth]{%
  \def\svgwidth{#1}
  \filemodCmp{#2.pdf}{#2.svg}{%}{%
    \immediate\write18{%
      inkscape -z -D --file=#2.svg --export-pdf=#2.pdf --export-latex
    }%
  }%
  \input{#2.pdf_tex}%
}
```

Mit `\immediate\write18{<externer Aufruf>}` wird das zwischenzeitliche Ausführen eines externen Programms beim Durchlauf von pdf \LaTeX – in diesem Fall von *inkscape.exe* – möglich. Damit der externe Aufruf auch tatsächlich durchgeführt wird, ist die Ausführung von pdf \LaTeX mit der Option `--shell-escape` beziehungsweise bei der Verwendung von *MiKTeX* mit `--enable-write18` zwingend notwendig. Außerdem muss der Pfad zur Datei *inkscape.exe* dem System bekannt sein.⁷⁷ Bei der Verwendung des Befehls `\includesvg` muss der Dateiname *ohne Endung* angegeben werden. Die einzubindende SVG-Datei sollte sich hierbei im gleichen Pfad wie das Hauptdokument befinden. Ist die SVG-Datei in einem Unterordner relativ zum Pfad des Hauptdokumentes, kann dieser einfach mit `\includesvg{<Ordner>/<Datei>}` im Argument angegeben werden.

7.22 FEHLERMELDUNG: ! NO ROOM FOR A NEW \WRITE

v2.02

Für das Erstellen und Schreiben externer Hilfsdateien steht \LaTeX 2 ϵ nur eine begrenzte Anzahl sogenannter Ausgabe-Streams zur Verfügung. Allein für jedes zu erstellende Verzeichnis reserviert \LaTeX selbst jeweils einen neuen Stream. Auch einige bereits zuvor in diesem Handbuch vorgestellte, sehr hilfreiche Pakete – wie beispielsweise [hyperref](#), [biblatex](#), [glossaries](#), [todonotes](#) oder auch [filecontents](#) – benötigen eigene Hilfsdateien und öffnen für das Erstellen dieser einen Ausgabe-Stream oder mehr. Lädt der Anwender mehrere, in eine Hilfsdatei schreibende Pakete, kann es zur Fehlermeldung

```
! No room for a new \write .
```

⁷⁶<http://www.ctan.org/pkg/svg-inkscape>

⁷⁷Der Pfad zu *inkscape.exe* in der Umgebungsvariable PATH des Betriebssystems enthalten sein.

kommen. Abhilfe schafft hier das Paket `scrwfile`, welches einige Änderungen am \LaTeX -Kern vornimmt, um die Anzahl der benötigten Hilfsdateien für das Schreiben aller Verzeichnisse zu reduzieren. Es muss einfach in der Präambel eingebunden werden. Sollten mit diesem Paket unerwarteter Weise Probleme auftreten, ist dessen Anleitung im [KOMA-Script-Handbuch](#)⁷⁸ zu finden. Eine weitere Möglichkeit, das beschriebene Problem der geringen Menge an Ausgabe-Streams zu umgehen, stellt das Paket `morewrites` dar. Allerdings ist dessen Verwendung nicht in allen Fällen von Erfolg gekrönt.

7.23 FEHLERMELDUNG BEIM LADEN EINES PAKETES MIT OPTIONEN

v2.05

Es kann unter Umständen passieren, dass beim Laden eines Paketes mit bestimmten Optionen via `\usepackage[<Option>]{<Paket>}` folgender Fehler ausgegeben wird:

```
! LaTeX Error: Option clash for package <...>.
```

Mit großer Sicherheit wird das angeforderte Paket bereits durch die verwendete Dokumentklasse oder ein anderes Paket geladen. Im einfachsten Fall genügt es, bereits vor dem Laden der Dokumentklasse mit `\documentclass` durch `\PassOptionsToPackage{<Optionen>}{<Paket>}` die gewünschten Optionen an das Paket weiterzureichen.

7.24 PROBLEME BEI DER VERWENDUNG VON AUTO-PST-PDF

v2.02

Bei der Verwendung von pdf \LaTeX liest das Paket `auto-pst-pdf` die Präambel ein und erstellt anschließend über den PostScript-Pfad `latex → dvips → ps2pdf` eine PDF-Datei, welche lediglich alle in den vorhandenen `pspicture`-Umgebungen erstellten Grafiken enthält. Das Paket `ifpdf` stellt das Makro `\ifpdf` bereit, mit welchem unterschieden werden kann, ob pdf \LaTeX als Dokumentprozessor verwendet wird. Abhängig davon können unterschiedliche Quelltexte ausgeführt werden, was genutzt wird, um die nachfolgend beschriebenen Probleme zu beheben.

```
\usepackage{ifpdf}
```

DIE GLEICHZEITIGE VERWENDUNG VON FLOATROW

Das Paket `floatrow` stellt Befehle bereit, mit denen die Beschriftung von Gleitobjekten sehr bequem gesetzt werden können. Diese Setzen ihren Inhalt erst in einer Box, um deren Breite zu ermitteln und diese anschließend auszugeben. In Kombination mit `auto-pst-pdf` führt das zu einer doppelten Erstellung der gewünschten Abbildung. Um dies zu vermeiden, müssen die durch `floatrow` bereitgestellten Befehle „unschädlich“ gemacht werden. Die fraglichen Befehlen akzeptieren allerdings bis zu drei optionale Argumente vor den beiden obligatorischen, was für die Benutzerschnittstelle für die (Re-)Definition durch $\text{\LaTeX}_{2\epsilon}$ normalerweise nicht vorgesehen ist. Deshalb wird das Paket `xparse` geladen, mit welchem dies möglich wird. Genauer dazu ist der dazugehörigen Paketdokumentation zu entnehmen. Mit folgendem Quelltextauszug kann man die `floatrow`-Befehle zusammen mit der `pspicture`-Umgebung wie gewohnt verwenden.

```
\usepackage{floatrow}
\usepackage{xparse}
\ifpdf\else
  \RenewDocumentCommand{\fcapside}{ooo+m+m}{#4#5}
  \RenewDocumentCommand{\ttabbox}{ooo+m+m}{#4#5}
```

⁷⁸<http://mirrors.ctan.org/macros/latex/contrib/koma-script/doc/scrguide.pdf>

```
\RenewDocumentCommand{\ffigbox}{ooo+m+m}{#4#5}
\fi
```

DIE PARALLELE NUTZUNG VON TIKZ UND TODONOTES

Mit dem Paket [tikz](#) – und auch allen anderen Paketen die selbiges nutzen wie beispielsweise [todonotes](#) – gibt es in Verbindung mit [auto-pst-pdf](#) ebenfalls Probleme. Dies lässt sich lösen, indem man diese Pakete lediglich in der Präambel einbindet, wenn pdf \LaTeX aktiv ist.

```
\ifpdf
\usepackage{tikz}%...gegebenenfalls weitere auf tikz basierende Pakete
\fi
```

TEIL III

ANHANG

ANHANG A

WEITERFÜHRENDE INSTALLATIONSHINWEISE

Achtung!

*Im Folgenden werden unterschiedliche Varianten erläutert, wie die jeweils aktuelle Version von TUD-KOMA-Script genutzt werden kann, falls bereits eine frühere Variante als **lokale Nutzerinstallation** verwendet wird. Unabhängig davon, für welche Möglichkeit Sie sich entscheiden, ist eine abermalige Installation der PostScript-Schriften für eine fehlerfreie Verwendung von TUD-KOMA-Script zwingend notwendig, wenn Sie bisher TUD-KOMA-Script vor der Version v2.02 verwendet haben, da der Installationsprozess der Schriften des Corporate Designs für die Version v2.02 nochmals angepasst werden musste. Dieser Schritt ließ sich leider nicht vermeiden. Für zukünftige Versionen kann darauf hoffentlich verzichtet werden, sodass Aktualisierungen ausschließlich über CTAN eingespielt werden können. In [Unterkapitel 1.2](#) wird der Installationsprozess der PostScript-Schriften beschrieben.*

Bis zur Version v2.01 wurde TUD-KOMA-Script ausschließlich über das [TUD- \$\text{\LaTeX}\$ -Forum](#)⁷⁹ zur lokalen Nutzerinstallation angeboten. In erster Linie hat das historische Hintergründe und hängt mit der Entstehungsgeschichte von TUD-KOMA-Script zusammen.

Eine lokale Nutzerinstallation bietet mehr oder weniger genau einen Vorteil. Treten bei der Verwendung von TUD-KOMA-Script Probleme auf, können diese im Forum gemeldet und diskutiert werden. Ist für ein solches Problem tatsächlich eine Fehlerkorrektur respektive eine Aktualisierung von TUD-KOMA-Script nötig, kann diese schnell und unkompliziert über das [GitHub-Repository tudscr](#)⁸⁰ bereitgestellt und durch den Anwender sofort genutzt werden.

Dies hat allerdings für Anwender, welche das Forum relativ wenig oder gar nicht besuchen, den großen Nachteil, dass diese nicht von Aktualisierungen, Verbesserungen und Fehlerkorrekturen neuer Versionen profitieren können. Auch sämtliche nachfolgenden Bugfixes und Aktualisierungen des TUD-KOMA-Script-Bundles müssen durch den Anwender manuell durchgeführt werden. Daher wird in Zukunft die Verbreitung via [CTAN](#)⁸¹ präferiert, so dass TUD-KOMA-Script stets in der aktuellen Version verfügbar ist – eine durch den Anwender aktuell gehaltene \LaTeX -Distribution vorausgesetzt. Der einzige Nachteil bei diesem Ansatz ist, dass die Verbreitung eines Bugfixes über das [Comprehensive TeX Archive Network \(CTAN\)](#)⁸² und die anschließende Bereitstellung durch die verwendete Distribution für gewöhnlich mehrere Tage dauert.

Die gängigen \LaTeX -Distributionen durchsuchen im Regelfall zuerst das lokale `texmf`-Nutzerverzeichnis nach Klassen und Paketen und erst daran anschließend den `texmf`-Pfad der Distribution selbst. Dabei spielt es keine Rolle, in welchem Pfad die neuere Version einer Klasse oder eines Paketes liegt. Sobald im Nutzerverzeichnis die gesuchte Datei gefunden wurde, wird die Suche beendet. *In der Konsequenz bedeutet dies, dass sämtliche Aktualisierungen über CTAN nicht zum Tragen kommen, falls TUD-KOMA-Script als lokale Nutzerversion installiert wurde.*

Achtung!

Deshalb wird Anwendern, die TUD-KOMA-Script in der Version v2.01 oder älter nutzen und sich nicht *bewusst* für eine lokale Nutzerinstallation entschieden haben, empfohlen, diese zu deinstallieren. Der Prozess der Deinstallation wird in [Unterkapitel A.1](#) erläutert. Wird diese einmalig durchgeführt, können Updates des TUD-KOMA-Script-Bundles durch die Aktualisierungsfunktion der Distribution erfolgen. Wie das TUD-KOMA-Script-Bundle trotzdem als lokale Nutzerversion installiert oder aktualisiert werden kann, ist in [Unterkapitel A.2](#) beziehungsweise [Unterkapitel A.3](#) zu finden. Der Anwender sollte in diesem Fall allerdings genau wissen, was er damit bezweckt, da er in diesem Fall für die Aktualisierung von TUD-KOMA-Script selbst verantwortlich ist.

⁷⁹<http://latex.wcms-file3.tu-dresden.de/phpBB3/>

⁸⁰<https://github.com/tud-cd/tudscr/releases>

⁸¹<http://www.ctan.org/pkg/tudscr>

⁸²<http://www.ctan.org/>

A.1 LOKALE DEINSTALLATION DES TUD-KOMA-SCRIPT-BUNDLES

Um die lokale Nutzerinstallation zu entfernen, kann für Windows [tudscr_uninstall.bat](#)⁸³ sowie für unixartige Betriebssysteme [tudscr_uninstall.sh](#)⁸⁴ verwendet werden. Nach der Ausführung des jeweiligen Skriptes kann in der Konsole beziehungsweise im Terminal mit

```
kpsewhich -all tudscrbase.sty
```

überprüft werden, ob die Deinstallation erfolgreich war oder immer noch eine lokale Nutzerinstallation vorhanden ist. Es werden alle Pfade ausgegeben, in welchen die Datei **tudscrbase.sty** gefunden wird. Erscheint nur noch der Pfad der Distribution, ist die TUD-KOMA-Script-Version von CTAN aktiv und der Anwender kann mit dem TUD-KOMA-Script-Bundle arbeiten. Falls es nicht schon passiert ist, müssen dafür lediglich die Schriften des Corporate Designs installiert werden ([Unterkapitel 1.2](#)).

Wird *nur* das lokale Nutzerverzeichnis oder gar kein Verzeichnis gefunden, so wird höchstwahrscheinlich eine veraltete Distribution verwendet. In diesem Fall wird eine Aktualisierung dieser *unbedingt* empfohlen. Sollte dies nicht möglich sein, *muss* TUD-KOMA-Script als lokale Nutzerversion aktualisiert ([Unterkapitel A.3](#)) beziehungsweise bei der erstmaligen Verwendung installiert ([Unterkapitel A.2](#)) werden. Sollte neben dem Pfad der Distribution immer noch mindestens ein weiterer Pfad angezeigt werden, so ist weiterhin eine lokale Nutzerversion installiert. In diesem Fall hat der Anwender zwei Möglichkeiten:

1. Entfernen der lokalen Nutzerinstallation (manuell)
2. Aktualisierung der lokalen Nutzerversion

Die erste Variante wird nachfolgend erläutert, die zweite Möglichkeit wird in [Unterkapitel A.3](#) beschrieben. Nur die manuelle Deinstallation der lokalen Nutzerversion TUD-KOMA-Script ermöglicht dabei die Verwendung der jeweils aktuellen CTAN-Version. Hierfür ist etwas Handarbeit durch den Anwender vonnöten. Der in der Konsole beziehungsweise im Terminal mit

```
kpsewhich -all tudscrbase.sty
```

gefundene – zum Ordner der Distribution *zusätzliche* – Pfad hat die folgende Struktur:

```
<Installationspfad>/tex/latex/tudscr/tudscrbase.sty
```

Um die Nutzerinstallation vollständig zu entfernen, muss als erstes zu *<Installationspfad>* navigiert werden. Anschließend ist in diesem Pfad Folgendes durchzuführen:

tex/latex/tudscr/	alle .cls- und .sty-Dateien löschen
tex/latex/tudscr/	Ordner logo vollständig löschen
doc/latex/	Ordner tudscr vollständig löschen
source/latex/	Ordner tudscr vollständig löschen

Das Verzeichnis *<Installationspfad>/tex/latex/tudscr/fonts* **sollte erhalten bleiben**. Andernfalls müssen die Schriften des Corporate Designs abermals wie unter [Unterkapitel 1.2](#) beschrieben installiert werden. Zum Abschluss ist in der Kommandozeile beziehungsweise im Terminal der Befehl `texhash` aufzurufen. Damit wurde die lokale Nutzerversion entfernt und es wird von nun an die Version von TUD-KOMA-Script genutzt, welche durch die verwendete Distribution bereitgestellt wird.

⁸³https://github.com/tud-cd/tudscr/releases/download/uninstall/tudscr_uninstall.bat

⁸⁴https://github.com/tud-cd/tudscr/releases/download/uninstall/tudscr_uninstall.sh

A.2 LOKALE INSTALLATION DES TUD-KOMA-SCRIPT-BUNDLES

Für die lokale Nutzerinstallation von TUD-KOMA-Script inklusive der Schriften des Corporate Designs werden für Windows sowie unixartige Betriebssysteme die passenden Skripte angeboten. Eine lokale Installation sollte nur von Anwender ausgeführt werden, die genau wissen, aus welchen Gründen dies geschehen soll.

A.2.1 LOKALE INSTALLATION VON TUD-KOMA-SCRIPT UNTER WINDOWS

Für eine lokale Nutzerinstallation des TUD-KOMA-Script-Bundles und der dazugehörigen Schriften für die Distributionen **T_EX Live** oder **MiK_TE_X** werden neben den Schriftarchiven die Dateien aus **TUD-KOMA-Script_v2.05_Windows_full.zip**⁸⁵ benötigt. Vor der Verwendung des Skripts **tudscr_v2.05_install.bat** sollte sichergestellt werden, dass sich *alle* der folgenden Dateien im selben Verzeichnis befinden:

tudscr_v2.05.zip	Archiv mit Klassen- und Paketdateien
tudscr_v2.05_install.bat	Installationsskript
Univers_PS.zip	Archiv mit Schriftdateien für Univers
DIN_Bd_PS.zip	Archiv mit Schriftdateien für DIN BOLD
tudscrfonts.zip	Archiv mit Metriken für die Schriftinstallation via fontinst

Beim Ausführen des Installationsskripts werden alle Schriften in das lokale Nutzerverzeichnis der jeweiligen Distribution installiert, falls kein anderes Verzeichnis explizit angegeben wird. Für Hinweise bei Problemen mit der Schriftinstallation sei auf [Abschnitt 1.2.1](#) verwiesen. *Wird die L^AT_EX-Distribution MiK_TE_X genutzt, sollte in jedem Fall vor der Ausführung der Installationsskripte ein Update der Distribution durchgeführt werden. Andernfalls wird es unter Umständen im Installationsprozess oder bei der Nutzung von TUD-KOMA-Script zu Problemen kommen.*

Achtung!

A.2.2 LOKALE INSTALLATION VON TUD-KOMA-SCRIPT UNTER LINUX UND OS X

Für eine lokale Nutzerinstallation des TUD-KOMA-Script-Bundles und der dazugehörigen Schriften für die Distributionen **T_EX Live** oder **MacT_EX** werden neben den Schriftarchiven die Dateien aus **TUD-KOMA-Script_v2.05_Unix_full.zip**⁸⁶ benötigt. Vor der Verwendung des Skripts **tudscr_v2.05_install.sh** sollte sichergestellt werden, dass sich *alle* der folgenden Dateien im selben Verzeichnis befinden:

tudscr_v2.05.zip	Archiv mit Klassen- und Paketdateien
tudscr_v2.05_install.sh	Installationsskript (Terminal: <code>bash tudscr_v2.05_install.sh</code>)
Univers_PS.zip	Archiv mit Schriftdateien für Univers
DIN_Bd_PS.zip	Archiv mit Schriftdateien für DIN BOLD
tudscrfonts.zip	Archiv mit Metriken für die Schriftinstallation via fontinst

Beim Ausführen des Installationsskripts werden alle Schriften in das lokale Nutzerverzeichnis der jeweiligen Distribution installiert. Für Hinweise bei Problemen mit der Schriftinstallation sei auf [Abschnitt 1.2.2](#) verwiesen.

⁸⁵https://github.com/tud-cd/tudscr/releases/download/v2.05/TUD-KOMA-Script_v2.05_Windows_full.zip

⁸⁶https://github.com/tud-cd/tudscr/releases/download/v2.05/TUD-KOMA-Script_v2.05_Unix_full.zip

A.3 LOKALES UPDATE DES TUD-KOMA-SCRIPT-BUNDLES

A.3.1 UPDATE DES TUD-KOMA-SCRIPT-BUNDLES AB VERSION v2.02

Für eine lokale Aktualisierung von TUD-KOMA-Script auf v2.05 muss das Archiv [TUD-KOMA-Script_v2.05_Windows_update.zip](#)⁸⁷ respektive [TUD-KOMA-Script_v2.05_Unix_update.zip](#)⁸⁸ entpackt und anschließend `tudscr_v2.05_update.bat` oder `tudscr_v2.05_update.sh` ausgeführt werden. *Die lokale Aktualisierung funktioniert nur, wenn TUD-KOMA-Script bereits mindestens in der Version v2.02 entweder als lokale Nutzerversion oder über die Distribution sowie die PostScript-Schriften installiert sind.*

Achtung!

A.3.2 UPDATE DES TUD-KOMA-SCRIPT-BUNDLES AB VERSION v2.00

Mit der Version v2.02 gab es einige tiefgreifende Änderungen. Deshalb wird für vorausgehende Versionen – sprich v2.00 und v2.01 – kein dediziertes Update angeboten. Die Aktualisierung kann durch den Anwender entweder – wie in [Unterkapitel A.2](#) erläutert – mit einer skriptbasierten oder mit einer manuellen Neuinstallation erfolgen. Für die zweite Variante muss der Inhalt des Archivs [tudscr_v2.05.zip](#)⁸⁹ in das lokale `texmf`-Nutzerverzeichnis kopiert werden. Des Weiteren wurde die Installation der Schriften überarbeitet. Deshalb wird auch für diese eine Neuinstallation ([Unterkapitel 1.2](#)) sehr empfohlen.

A.3.3 UPDATE DES TUD-KOMA-SCRIPT-BUNDLES VON VERSION v1.0

Ist TUD-KOMA-Script in der veralteten Version v1.0 installiert, so wird dringend zu einer Deinstallation geraten. Andernfalls wird es zu Problemen kommen. Dafür werden die Skripte [tudscr_uninstall.bat](#)⁹⁰ respektive [tudscr_uninstall.sh](#)⁹¹ bereitgestellt. Die aktuelle Version v2.05 kann nach der Deinstallation wie in [Unterkapitel 1.2](#) beschrieben installiert werden.

Möchten Sie die obsoleten TUD-KOMA-Script-Klassen in der Version v1.0 nach einer Aktualisierung weiterhin nutzen, so müssen diese nach der Deinstallation neu installiert werden. Dafür steht das Archiv [TUD-KOMA-Script_v1.0old.zip](#)⁹² bereit, welches sowohl die genannten Skripte zur Deinstallation als auch die zur neuerlichen Installation der veralteten Klassen benötigten `tudscr_v1.0old_install.bat` oder `tudscr_v1.0old_install.sh` enthält. Nach Abschluss des Vorgangs sind die alten Klassen der Version v1.0 mit `tudscrbookold`, `tudscrreptold` und `tudscrartclold` parallel zur aktuellen Version v2.05 verwendbar.

Im Vergleich zur Version v1.0 hat sich an der Benutzerschnittstelle nicht sehr viel verändert. Treten nach dem Umstieg auf die Version v2.05 dennoch Probleme auf, sollte der Anwender als erstes die Beschreibung des Paketes `tudscrcomp` in [Unterkapitel B.2 auf Seite 127](#) lesen, welches eine Schnittstelle zur Nutzung alter und ursprünglich nicht mehr vorgesehener Befehle sowie Optionen bereitstellt. Allerdings werden einige von diesen auch durch das Paket `tudscrcomp` nicht mehr bereitgestellt. Aufgeführt sind diese in [Anhang B](#). Sollten trotz aller Hinweise dennoch Fehler oder Probleme beim Umstieg auf die neue TUD-KOMA-Script-Version auftreten, ist eine Meldung im [TUD- \$\LaTeX\$ -Forum](#)⁹³ die beste Möglichkeit, um Hilfe zu erhalten.

⁸⁷https://github.com/tud-cd/tudscr/releases/download/v2.05/TUD-KOMA-Script_v2.05_Windows_update.zip

⁸⁸https://github.com/tud-cd/tudscr/releases/download/v2.05/TUD-KOMA-Script_v2.05_Unix_update.zip

⁸⁹https://github.com/tud-cd/tudscr/releases/download/v2.05/tudscr_v2.05.zip

⁹⁰https://github.com/tud-cd/tudscr/releases/download/uninstall/tudscr_uninstall.bat

⁹¹https://github.com/tud-cd/tudscr/releases/download/uninstall/tudscr_uninstall.sh

⁹²https://github.com/tud-cd/tudscrold/releases/download/v1.0/TUD-KOMA-Script_v1.0old.zip

⁹³<http://latex.wcms-file3.tu-dresden.de/phpBB3/>

A.4 INSTALLATIONSHINWEISE FÜR PORTABLE INSTALLATIONEN

Prinzipiell ist die Installation der PostScript-Schriften bei der Nutzung von **T_EX Live Portable** beziehungsweise **MiK_TE_X Portable** äquivalent zur nicht-portablen Variante, welche in [Unterkapitel 1.2](#) beschrieben wird. Alle dort gegebenen Hinweise sollten sorgfältig berücksichtigt werden. Darüberhinaus ist bei **MiK_TE_X Portable** darauf zu achten, den Installationspfad nicht unbedingt auf der obersten Verzeichnisebene des externen Speichermediums zu wählen. Zur Installation der Schriften des Corporate Designs wird das Archiv **TUD-KOMA-Script_fonts_Windows.zip**⁹⁴ benötigt. Dieses kann entweder auf der lokalen Festplatte oder auf dem externen Speichermedium entpackt werden. Danach wird folgendes Vorgehen empfohlen:

T_EX LIVE PORTABLE

Das folgende Vorgehen wurde mit Windows getestet. Empfehlungen für die portable Installation für unixoide Betriebssysteme können gerne an tudscr@gmail.com gesendet werden.

1. Installation von **T_EX Live Portable** in den Pfad `<Laufwerk>:\texlive`
2. Die Datei **tl-tray-menu.exe** im Installationspfad öffnen
3. Im Infobereich der Taskleiste mit einem Rechtsklick auf das Symbol von **T_EX Live Portable** das Kontextmenü öffnen und ein Update entweder über die grafische Oberfläche (*Package Manager*) oder die Kommandozeile (*Command Prompt*) durchführen
4. Über das Kontextmenü die Kommandozeile starten und über diese das Installationsskript für die Schriften **tudscrfonts_install.bat** ausführen. Der voreingestellte Installationspfad kann im Normalfall so belassen werden. Wird dieser geändert, so sollte dieser sich logischerweise auf dem externen Speichermedium befinden. *Ein Ausführen ohne die über **T_EX Live Portable** geöffnete Kommandozeile führt zu Fehlern.*

Achtung!

MiK_TE_X PORTABLE

1. Installation von **MiK_TE_X Portable** in den Pfad `<Laufwerk>:\LaTeX\MiKTeXportable`
2. Die Datei **miktex-portable.cmd** im Installationspfad öffnen
3. Im Infobereich der Taskleiste mit einem Rechtsklick auf das Symbol von **MiK_TE_X Portable** das Kontextmenü öffnen und ein Update durchführen
4. Über das Kontextmenü die Kommandozeile starten und über diese das Installationsskript für die Schriften **tudscrfonts_install.bat** ausführen. Der voreingestellte Installationspfad kann im Normalfall so belassen werden. Wird dieser geändert, so sollte dieser sich logischerweise auf dem externen Speichermedium befinden. Bei diesem Schritt werden die Pakete **fontinst**, **cmbright** und **iwona** unter Umständen nachinstalliert. *Ein Ausführen ohne die über **MiK_TE_X Portable** geöffnete Kommandozeile führt zu Fehlern.*
5. Bei der erstmaligen Verwendung einer der TUD-KOMA-Script-Dokumentklassen werden die Pakete **tudscr**, **koma-script**, **etoolbox**, **textcase**, **environ**, **trimspaces**, **xcolor**, **mptopdf** durch **MiK_TE_X Portable** nachinstalliert, falls diese nicht schon vorhanden sind und die automatische Nachinstallation von Paketen aktiviert ist.

Achtung!

⁹⁴https://github.com/tud-cd/tudscr/releases/download/fonts/TUD-KOMA-Script_fonts_Windows.zip

ANHANG B

OBSOLETE SOWIE VOLLSTÄNDIG ENTFERNTE OPTIONEN UND BEFEHLE

B.1 VERALTETE OPTIONEN UND BEFEHLE IN TUD-KOMA-SCRIPT

Einige Optionen und Befehle waren während der Weiterentwicklung von TUD-KOMA-Script in ihrer ursprünglichen Form nicht mehr umsetzbar oder wurden – unter anderem aus Gründen der Kompatibilität zu anderen Paketen – schlichtweg verworfen. Dennoch besteht für die meisten entfallenen Direktiven eine Möglichkeit, deren Funktionalität ohne größere Aufwände mit TUD-KOMA-Script in der aktuellen Version v2.05 darzustellen. Ist dies der Fall, wird hier entsprechend kurz darauf hingewiesen.

B.1.1 ÄNDERUNGEN FÜR TUD-KOMA-SCRIPT V2.00

<code>cd=alternative</code>	(<i>entfällt</i>)
<code>cdtitle=alternative</code>	(<i>entfällt</i>)
<code>\titlecolwidth</code> (Länge)	(<i>entfällt</i>)
<code>\authortext</code> (Bezeichner)	(<i>entfällt</i>)

Die alternative Titelseite ist komplett aus dem TUD-KOMA-Script-Bundle entfernt worden. Dementsprechend entfallen auch die dazugehörigen Optionen sowie Länge und Bezeichner.

<code>color=<Ein-Aus-Wert></code>

 (siehe [cd](#) auf dieser Seite)

Die Einstellungen der farbigen Ausprägung des Dokumentes erfolgt über die Option [cd](#).

<code>tudfonts=<Ein-Aus-Wert></code>
<code>cdfonts=<Ein-Aus-Wert></code>

 (siehe [cdfont](#) auf Seite [17](#))

Die Option zur Schrifteinstellung ist wesentlich erweitert worden. Aus Gründen der Konsistenz wurde diese umbenannt.

<code>tudfoot=<Ein-Aus-Wert></code>

 (siehe [cdfont](#) auf Seite [26](#))

Ebenso wurde die Option [tudfoot](#) umbenannt, um dem Namensschema der restlichen Optionen von TUD-KOMA-Script zu entsprechen.

<code>headfoot=<Einstellung></code>

 (*entfällt*, KOMA-Script-Optionen `headinclude` und `footinclude`)

Diese Option war für TUD-KOMA-Script in der Version v1.0 notwendig, um die parallele Verwendung der beiden Pakete [typearea](#) und [geometry](#) zu ermöglichen. Die Erstellung des Satzspiegels wurde komplett überarbeitet. Mittlerweile werden an das Paket [geometry](#) die Einstellungen für die KOMA-Script-Optionen `headinclude` und `footinclude` direkt weitergereicht, so dass die Option [headfoot](#) nicht mehr notwendig ist und deshalb entfernt wurde.

<code>partclear=<Ein-Aus-Wert></code>
<code>chapterclear=<Ein-Aus-Wert></code>

 (*entfällt*, siehe [cleardoublespecialpage](#) auf Seite [37](#))

Beide Optionen sind in der neuen Option [cleardoublespecialpage](#) aufgegangen, womit ein konsistentes Layout erreicht wird. Die ursprünglichen Optionen entfallen.

<code>abstracttotoc=<Ein-Aus-Wert></code>
<code>abstractdouble=<Ein-Aus-Wert></code>

 (*entfällt*, siehe [abstract](#) auf Seite [40](#))

Beide Optionen wurden in die Option [abstract](#) integriert und sind deshalb überflüssig.

<code>\logfile{<Dateiname>}</code>	(siehe \headlogo auf Seite 27)
--	--

Der Befehl `\logfile` wurde in `\headlogo` umbenannt, wobei die Funktionalität weiterhin bestehen bleibt.

<code>bookmarks=<Ein-Aus-Wert></code>	(siehe tudbookmarks auf Seite 48)
---	---

Umbenannt, um Überschneidungen mit [hyperref](#) zu vermeiden.

<code>\signatureheight</code> (Länge)	(entfällt)
---------------------------------------	------------

Die Höhe für die Zeile der Unterschriften wurde dehnbar gestaltet, eine etwaige Anpassung durch den Anwender ist nicht vonnöten.

<code>\titlecoldelim</code> (Bezeichner)	(entfällt, siehe \titledelimiter auf Seite 32)
--	--

Das Trennzeichen für Bezeichnungen beziehungsweise beschreibende Texte und dem eigentlichen Feld auf der Titelseite ist nicht mehr sprachabhängig und wurde umbenannt.

<code>\confirmationandrestriction</code>	(entfällt, siehe \declaration auf Seite 47)
<code>\restrictionandconfirmation</code>	(entfällt, siehe \declaration auf Seite 47)

Die beiden Befehle entfallen, stattdessen sollte entweder der Befehl `\declaration` oder die Umgebung `declarations` zusammen mit den Befehlen `\confirmation` und `\blocking` verwendet werden, wobei sich diese in der Umgebung in beliebiger Reihenfolge anordnen lassen.

<code>\location{<Ort>}</code>	(siehe \place auf Seite 47)
-------------------------------------	---

In Anlehnung an andere L^AT_EX-Pakete und -Klassen wurde `\location` in `\place` umbenannt.

AUFGABENSTELLUNG

Die Umgebung für die Erstellung einer Aufgabenstellung für eine wissenschaftliche Arbeit wurde in das Paket `tudscrsupervisor` ausgelagert. Dieses muss für die Verwendung der Umgebung `task` und der daraus abgeleiteten standardisierten Form zwingend geladen werden.

<code>cdtask=<Einstellung></code>	(entfällt, siehe task auf Seite 57)
<code>taskcompact=<Ein-Aus-Wert></code>	(entfällt, siehe task auf Seite 57)
<code>\taskcolwidth</code> (Länge)	(entfällt, siehe task auf Seite 57)

Die Klassenoption `cdtask` ist komplett entfernt worden, alle Einstellungen, welche `task` betreffen erfolgen direkt über das optionale Argument der Umgebung. Die Variante eines kompakten Kopfes mit der Option `taskcompact` wird nicht mehr bereitgestellt. Die manuelle Einstellmöglichkeit der Spaltenbreite für den Kopf der Aufgabenstellung mit `\taskcolwidth` wurde aufgrund der verbesserten automatischen Berechnung entfernt.

<code>\tasks{<Ziele>}{<Schwerpunkte>}</code>	(entfällt, siehe \taskform auf Seite 57)
<code>\focustext</code> (Bezeichner)	(entfällt, siehe \focusname auf Seite 61)
<code>\objectivestext</code> (Bezeichner)	(entfällt, siehe \objectivesname auf Seite 61)

Der Befehl `\tasks` wurde in `\taskform` umbenannt und in der Funktionalität erweitert. Die darin verwendeten Bezeichner wurden ebenfalls leicht abgewandelt.

<code>\studentid{<Matrikelnummer>}</code>	(siehe \matriculationnumber auf Seite 32)
<code>\enrolmentyear{<Immatrikulationsjahr>}</code>	(siehe \matriculationyear auf Seite 32)
<code>\submissiondate{<Datum>}</code>	(siehe \date auf Seite 34)
<code>\birthday{<Geburtsdatum>}</code>	(siehe \dateofbirth auf Seite 32)
<code>\birthplace{<Geburtsort>}</code>	(siehe \placeofbirth auf Seite 32)
<code>\startdate{<Ausgabedatum>}</code>	(entfällt, siehe \issuedate auf Seite 58)

Alle Befehle wurden umbenannt und sind jetzt neben der Aufgabenstellung auch für die Titelseite im Corporate Design nutzbar.

<code>\studentidname</code> (Bezeichner)	(entfällt, siehe \matriculationnumbername auf Seite 50)
<code>\enrolmentname</code> (Bezeichner)	(entfällt, siehe \matriculationyearname auf Seite 50)
<code>\submissiontext</code> (Bezeichner)	(entfällt, siehe \datetext auf Seite 50)
<code>\birthdaytext</code> (Bezeichner)	(entfällt, siehe \dateofbirthtext auf Seite 50)
<code>\birthplacetext</code> (Bezeichner)	(entfällt, siehe \placeofbirthtext auf Seite 50)
<code>\supervisorIIname</code>	(entfällt, siehe \supervisorothername auf Seite 49)
<code>\defensetext</code> (Bezeichner)	(entfällt, siehe \defensedatetext auf Seite 50)
<code>\starttext</code> (Bezeichner)	(entfällt, siehe \issuedatetext auf Seite 61)
<code>\duetext</code> (Bezeichner)	(entfällt, siehe \duedatetext auf Seite 61)

Die Bezeichner wurden in Anlehnung an die dazugehörigen Befehlsnamen umbenannt.

B.1.2 ÄNDERUNGEN FÜR TUD-KOMA-SCRIPT V2.02

<code>\chapterheadingvskip</code> (Länge)	(entfällt, siehe \pageheadingsvskip auf Seite 23)
---	---

Die vertikale Positionierung von Überschriften wurde aufgeteilt. Zum einen kann diese für Titel-, Teile- und Kapitelseiten ([chapterpage=true](#)) über die Länge [\pageheadingsvskip](#) geändert werden. Für Kapitelüberschriften ([chapterpage=false](#)) sowie den Titelpf (titlepage=false) kann dies unabhängig davon mit [\headingsvskip](#) erfolgen.

<code>\degree[<Abk.>]{<Grad>}</code>	(siehe \graduation auf Seite 34)
<code>\degreertext</code> (Bezeichner)	(siehe \graduationtext auf Seite 50)

Der Befehl wurde zur Erhöhung der Kompatibilität mit anderen Paketen umbenannt, der dazugehörige Bezeichner dahingehend angepasst.

<code>\restriction[<Firma> <Parameterliste>]</code>	(siehe \blocking auf Seite 47)
<code>\restrictionname</code> (Bezeichner)	(siehe \blockingname auf Seite 51)
<code>\restrictiontext</code> (Bezeichner)	(siehe \blockingtext auf Seite 51)

Der Befehl wurde zur Erhöhung der Kompatibilität mit anderen Paketen umbenannt, die dazugehörigen Bezeichner dahingehend angepasst.

ÄNDERUNGEN IM PAKET TUDSCRSUPERVISOR

Im Paket [tudscrsupervisor](#) gab es ein paar kleinere Anpassungen.

<code>\branch{<Studienrichtung>}</code>	(entfällt, siehe \discipline auf Seite 58)
<code>\branchname</code> (Bezeichner)	(entfällt, siehe \disciplinename auf Seite 61)

Für die Aufgabenstellung wurden der Befehl sowie der dazugehörige Bezeichner umbenannt.

<code>\contact{<Kontaktperson(en)>}</code>	(entfällt, siehe <code>\contactperson</code> auf Seite 60)
<code>\contactname</code> (Bezeichner)	(entfällt, siehe <code>\contactpersonname</code> auf Seite 62)
<code>\phone{<Telefonnummer>}</code>	(entfällt, siehe <code>\telephone</code> auf Seite 60)
<code>\email{<E-Mail-Adresse>}</code>	(entfällt, siehe <code>\emailaddress</code> auf Seite 60)

Alle genannten Befehle und Bezeichner wurden für den Aushang umbenannt.

B.1.3 ÄNDERUNGEN FÜR TUD-KOMA-SCRIPT V2.03

<code>geometry=<Ein-Aus-Wert></code>	(siehe <code>cdgeometry</code> auf Seite 38)
--	--

Die Option `geometry` wurde aus Gründen der Konsistenz und dem Vermeiden eines möglichen Konfliktes mit einer späteren KOMA-Script-Version umbenannt. An der Funktionalität wurde nichts geändert.

<code>sansmath=<Ein-Aus-Wert></code>	(siehe <code>cdmath</code> auf Seite 20)
--	--

Die Option `sansmath` wurde aus Gründen der Konsistenz umbenannt. Zusätzlich wurde die Funktionalität erweitert.

<code>barfont=<Einstellung></code>	(siehe <code>cdhead</code> auf Seite 25)
<code>widehead=<Ein-Aus-Wert></code>	(siehe <code>cdhead</code> auf Seite 25)

Die Funktionalitäten der Optionen `barfont` und `widehead` wurden in der Option `cdhead` zusammengefasst und erweitert.

B.1.4 ÄNDERUNGEN FÜR TUD-KOMA-SCRIPT V2.04

<code>fontspec=<Ein-Aus-Wert></code>	<code>preset: false</code>
--	----------------------------

Anstatt die Option `fontspec` zu aktivieren, kann einfach das Paket `fontspec` in der Dokumentpräambel geladen werden. Dadurch können anschließend zusätzliche Pakete genutzt werden, die auf die Verwendung von `fontspec` angewiesen sind. Sollte die Option `fontspec` dennoch genutzt werden, müssen alle auf das Paket `fontspec` aufbauende Einstellungen mit `\AfterPackage{fontspec}{...}` durch den Anwender verzögert werden. In [Abschnitt 2.1.3 auf Seite 21](#) sind weitere Hinweise zur Verwendung des Paketes `fontspec` zu finden.

false/off/no: (Säumniswert)

Die Hausschriften im Stil des Corporate Designs der Technischen Universität Dresden werden im PostScript-Format eingebunden. Sowohl Kerning als auch der mathematische Satz funktionieren problemlos.

true/on/yes: (Säumniswert)

Es werden die OpenType-Varianten der Hausschriften verwendet. Dazu wird das Paket `fontspec` geladen, welches lediglich mit Lua \TeX oder Xe \LaTeX jedoch nicht mit pdf \LaTeX als genutzt werden kann. Sowohl beim mathematischen Satz als auch beim Kerning der Schriften kann es zu Problemen kommen. Hierfür müssen die OpenType-Schriften auf dem Betriebssystem installiert sein. Die Verwendung dieser Einstellung sollte nur für den Fall erfolgen, dass eine Installation der PostScript-Schriften nicht möglich ist.

B.2 DAS PAKET TUDSCRCOMP – UMSTIEG VON ANDEREN KLASSEN

Achtung!

Sollten Sie eine der Klassen `tudbook`, `tudbeamer`, `tudletter`, `tudfax`, `tudhaus` und `tudform` oder TUD-KOMA-Script in der Version v1.0 nie genutzt haben, können Sie dieses Unterkapitel ohne Weiteres überspringen. Sämtliche hier vorgestellten Optionen und Befehle sind in der aktuellen Version von TUD-KOMA-Script obsolet.

Zu Beginn der Entwicklung von TUD-KOMA-Script bildete die Klasse `tudbook` die Basis. Ziel war es, sämtliche Funktionalitäten dieser beizubehalten und zusätzlich den vollen Funktionsumfang der KOMA-Script-Klassen nutzbar zu machen. Bei der kompletten Neuimplementierung der TUD-KOMA-Script-Klassen wurde sehr viel verändert und verbessert. Einige der Optionen und Befehle waren jedoch bereits in der TUD-KOMA-Script-Version v1.0 Relikte, um die Kompatibilität zur `tudbook`-Klasse und ihren Derivaten zu gewährleisten. Mit TUD-KOMA-Script in der Version v2.00 wurden einige aus Gründen der Konsistenz lediglich umbenannt, andere wiederum wurden vollständig entfernt oder über neue Befehle und Optionen in ihrer Funktionalität ersetzt und erweitert.

Das Paket `tudscrcomp` dient der Überführung von alten Dokumenten, die entweder mit der `tudbook`-Klasse, ihren Derivaten oder mit TUD-KOMA-Script in der Version v1.0 erstellt wurden, auf TUD-KOMA-Script v2.05. Es werden einige Optionen und Befehle bereitgestellt, welche von den alten Klassen definiert wurden und das entsprechende Verhalten nachahmen. Damit soll die Kompatibilität bei der Änderung der Dokumentklasse sichergestellt werden. Die Intention ist, alte Dokumente möglichst schnell und einfach auf die TUD-KOMA-Script-Klassen portieren zu können. Des Weiteren ist beschrieben, wie sich die Funktionalität ohne die Verwendung des Paketes `tudscrcomp` mit den Mitteln von TUD-KOMA-Script umsetzen lassen. Für den Satz neuer Dokumente wird empfohlen, auf den Einsatz dieses Paketes komplett zu verzichten und stattdessen die neuen Befehle zu nutzen.

<code>\einrichtung{<Fakultät>}</code>	(identisch zu <code>\faculty</code>)
<code>\fachrichtung{<Einrichtung>}</code>	(identisch zu <code>\department</code>)
<code>\institut{<Institute>}</code>	(identisch zu <code>\institute</code>)
<code>\professur{<Lehrstuhl>}</code>	(identisch zu <code>\chair</code>)

Dies sind die deutschsprachigen Befehle für den Kopf im Corporate Design.

<code>serifmath</code>	(identisch zu <code>cdmath=false</code>)
------------------------	---

Die Funktionalität wird durch die Option `cdmath` bereitgestellt.

<code>\tudfont{<Scriptart>}</code>	(identisch zu <code>\cdfont</code>)
--	--------------------------------------

Die direkte Auswahl der Schriftart sollte mit `\cdfont` erfolgen. Zusätzlich gibt es den Befehl `\textcdfont`, mit dem die Auszeichnung eines bestimmten Textes in einer anderen Schriftart erfolgen kann, ohne die Dokumentschrift umzuschalten.

B.2.1 OPTIONEN UND BEFEHLE AUS TUDBOOK & CO.

<code>colortitle</code>	(identisch zu <code>cdtitle=color</code>)
<code>nocolortitle</code>	(identisch zu <code>cdtitle=true</code>)

Die Funktionalität wird durch die Option `cdtitle` bereitgestellt.

`\moreauthor{<Autorenzusatz>}` (identisch zu `\authormore`)

Ursprünglich war diese Befehl für das Unterbringen aller möglichen, zusätzlichen Autoreninformationen gedacht. Auch der Befehl `\authormore` ist ein Rudiment davon. Empfohlen wird die Verwendung der Befehle `\dateofbirth`, `\placeofbirth`, `\matriculationnumber` und `\matriculationyear` sowie `\course` und `\discipline` aus dem Paket **tudscrsupervisor** für die Aufgabenstellung einer wissenschaftlichen Arbeit.

`\submitdate{<Datum>}` (identisch zu `\date`)

Die Funktionalität wird durch den erweiterten Standardbefehl `\date` abgedeckt.

`\supervisorII{<Name>}` (identisch zur Verwendung von `\and` innerhalb von `\supervisor`)

Es ist `\supervisor{<Name> \and <Name>}` statt `\supervisorII{<Name>}` zu verwenden.

<code>\supervisedby{<Bezeichnung>}</code>	(siehe <code>\supervisorname</code>)
<code>\supervisedIIby{<Bezeichnung>}</code>	(siehe <code>\supervisorothername</code>)
<code>\submittedon{<Bezeichnung>}</code>	(siehe <code>\datetext</code>)

Zur Änderung der Bezeichnung der Betreuer sollten die sprachabhängigen Bezeichner wie in [Unterkapitel 2.4](#) beschrieben angepasst werden. Eine Verwendung der alten Befehle entfernt die Abhängigkeit der Bezeichner von der verwendeten Sprache.

`ddcfooter` (identisch zu `ddcfoot=true`)

Die Funktionalität wird durch die Option `ddcfoot` bereitgestellt.

`\dissertation`

Die Funktionalität kann durch die Befehle `\thesis{diss}` und `\referee` sowie die Bezeichner `\refereename` und `\refereeothername` dargestellt werden.

`\chapterpage`

Durch diesen Befehl können Kapitelseiten konträr zur eigentlichen Einstellung aktiviert oder deaktiviert werden. Prinzipiell ist dies auch durch eine Änderung der Option `chapterpage` möglich. Allerdings wird davon abgeraten, da dies zu einem inkonsistenten Layout innerhalb des Dokumentes führt.

<pre>\begin{theglossary}[<Präambel>] ... \end{theglossary} \glossitem{<Begriff>}</pre>
--

Die **tudbook**-Klasse stellt eine rudimentäre Umgebung für ein Glossar bereit. Allerdings gibt es dafür bereits zahlreiche und besser implementierte Pakete. Daher wird für diese Umgebung keine Portierung vorgenommen, sondern lediglich die ursprüngliche Definition übernommen. Allerdings sein an dieser Stelle auf wesentlich bessere Lösungen wie beispielsweise das Paket **glossaries** oder – mit Abstrichen – das nicht ganz so umfangreiche Paket **nomenc** verwiesen.

ANHANG C

IDENTIFIKATION VON TUD-KOMA-SCRIPT

Im TUD-KOMA-Script-Bundle gibt es neben den Klassen selbst auch noch zusätzliche Pakete. Ein Teil dieser Pakete – genauer [tudscrsupervisor](#) und [tudscrcomp](#) – sind ausschließlich mit den TUD-KOMA-Script-Klassen nutzbar, andere wiederum – die beiden Pakete für Belange des Corporate Designs [tudscrfonts](#) (Schriften) und [tudscrcolor](#) (Farben) sowie die davon vollkommen unabhängigen Pakete [mathswap](#) und [twocolfix](#) – können mit allen existierenden $\text{\LaTeX}2_{\epsilon}$ -Dokumentklassen genutzt werden. Sämtliche Klassen und Pakete aus dem TUD-KOMA-Script-Bundle enthalten Befehle, welche diese als Bestandteil identifizieren.

v2.04

`\TUDScript`

Diese Anweisung setzt das Logo respektive die Wortmarke „TUD-KOMA-Script“ in serifenloser Schrift und mit leichter Sperrung des in Versalien gesetzten Teils. Dieser Befehl wird von allen Klassen und Paketen des TUD-KOMA-Script-Bundles mit `\DeclareRobustCommand`.

v2.04

`\TUDVersion`

In dieser Anweisung ist die Hauptversion von TUD-KOMA-Script in der Form

`<Datum> <Version> TUD-KOMA-Script`

abgelegt. Diese Version ist im TUD-KOMA-Script-Bundle für alle Klassen und Pakete gleich und kann daher nach dem Laden einer Klasse oder eines Paketes abgefragt werden. Diese Anleitung wurde beispielsweise mit „2015/08/26 v2.05 BETA TUD-KOMA-Script“ erstellt.

v2.04

`\TUDClassName`

In `\TUDClassName` ist der Name der aktuell verwendeten TUD-KOMA-Script-Klasse abgelegt. Will man also wissen, ob eine oder welche TUD-KOMA-Script-Klasse verwendet wird, so kann man einfach auf diese Anweisung testen. Von KOMA-Script werden zusätzlich noch die Anweisungen `\KOMAClassName` und `\ClassName` definiert, welche den Namen der zugrundeliegenden KOMA-Script-Klasse sowie die durch diese ersetzte Standardklasse enthalten.

INDEX

Die Formatierung der Einträge in allen aufgeführten Indizes ist folgendermaßen aufzufassen: **Zahlen in fetter Schrift** verweisen auf die **Erklärung** zu einem Stichwort, wobei in der digitalen Fassung dieses Handbuchs dieser Eintrag selbst ein Hyperlink zu seiner Erläuterung ist. Seitenzahlen in normaler Schriftstärke hingegen deuten auf zusätzliche Informationen, wobei diese für *kursiv hervorgehobene Zahlen* als besonders *wichtig* erachtet werden.

Bei Einträgen für **Klassen- und Paketoptionen** beziehungsweise für **Umgebungen und Befehle**, zu denen keine direkte **Erklärung** gegeben ist sondern lediglich zusätzliche Hinweise vorhanden sind, handelt es sich um KOMA-Script-Optionen. Diese sind gegebenenfalls im **dazugehörigen Handbuch** nachzulesen.

ALLGEMEINER INDEX

Die aufgelisteten Schlagworte sollen sowohl Antworten bei generellen Fragen als auch Lösungen für typische Probleme beim Umgang mit \LaTeX 2 ϵ sowie dem TUD-KOMA-Script-Bundle liefern.

A

Abbildungen *siehe* Grafiken
Abkürzungsverzeichnis 95
Absatzauszeichnung 107
Abschlussarbeit 33, 49–50
 Typisierung 33
Änderungen
 → [Änderungsliste](#) 146
Aktualisierung *siehe* Update
Aufgabenstellung 48, 57–58
Aufzählungen *siehe* Listen
Aushang 48, 59–60
Autorenangaben 32, 50
 kollaborativ 32

B

Befehle
 → [Index der Befehle etc.](#) 136
 Deklaration 89, 92, 100, 105
 Geltungsbereich 105
Berechnungen 99
Betreuer 34, 49
Bezeichner 33, 48–52, 60–62, 90
 → [Index der Bezeichner](#) 140
 Typisierung 33
biber (Anwendungssoftware) 95, 103
Bindekorrektur 39

C

Changelog
 → [Änderungsliste](#) 146
Citavi (Anwendungssoftware) 103
Cover *siehe* Umschlagseite

D

Dateien
 → [Index der Dateien etc.](#) 143
Datum 34, 46, 50, 89
 Geburtsdatum 32, 50
 Verteidigungsdatum 34, 50
Debugging 101
Deinstallation 118

DeLig (Anwendungssoftware) 92
Dezimaltrennzeichen .. *siehe* Trennzeichen
Distribution 9
 MacTeX *siehe* *MacTeX* (Distribution)
 MiKTeX *siehe* *MiKTeX* (Distribution)
 TeX Live *siehe* *TeX Live* (Distribution)
Drittlogo 27

E

Eclipse (Anwendungssoftware) 103
Eingabekodierung 90
Einheiten 94, 107

F

Fachreferent *siehe* Referent
Farben 40, 63, 89, 101
 → [Index der Farben](#) 142
 Farbmodell 65
Formelzeichenverzeichnis 95
Fußnoten 48, 107
Fußzeile 27–29, 39

G

Gleitobjekte 97, 110
 Beschriftung 97
 Platzierung 97, 110
Gliederungsebenen ... *siehe* Überschriften
Glossar 95
GNU Aspell (Anwendungssoftware) 92
Grafiken 88, 97, 98, 112
 Beschriftung *siehe* Gleitobjekte
Griechische Buchstaben 20
Großbuchstaben *siehe* Schriftauszeichnung
Gutachten 48, 58–59
Gutachter 34, 49

H

Hauptklassen 16
Hochschullehrer 49
Hunspell (Anwendungssoftware) 92

I

Inhaltsverzeichnis 48, 107

Inkscape (Anwendungssoftware) 98, 112
 Installation 10
 Nutzerinstallation 119
 Schriften 12

J

JabRef (Anwendungssoftware) 103

K

Kapitelseiten 23, 36, 37–38
 Kapitelüberschriften 23
Kile (Anwendungssoftware) 103
 Klassen
 → [Index der Dateien etc.](#) 143
 Klassenoptionen *siehe* Optionen
 Kleinbuchstaben *siehe* Schriftauszeichnung
 Kollaboratives Schreiben 32, 58
 Kolumnentitel 18, 26
 Kompatibilität 52
 Pakete 88
 tudbook 127–128
 Kopfzeile 25, 27, 29, 39
 Kurzfassung *siehe* Zusammenfassung

L

LaTeXDraw (Anwendungssoftware) 98
 Layout 22, 29
 Bindekorrektur 39
 Drittlogo 27
 Fußzeile 27–29, 39
 Kapitelseiten 23, 36, 37–38
 Kolumnentitel 18, 26
 Kopfzeile 25, 27, 29, 39
 Querbalken 25
 Rückseiten 37
 Satzspiegel 38–40
 Seitenränder 38
 Seitenstil 29, 38
 Teileseiten 23, 36–38
 Titel 22, 30, 38
 Überschriften 23
 Umschlagseite 23, 29
 Zweitlogo 27–28
LEd (Anwendungssoftware) 103
 Leerseiten *siehe* Vakantseiten
 Lesezeichen 48, 52, 89, 99
 Ligaturen 92
 Listen 95

Literaturverzeichnis 95, 105
 DRESDEN-concept-Logo 27–28
 Lokalisierung *siehe* Bezeichner
LyX (Anwendungssoftware) 103

M

MacTeX (Distribution) 9–12, 119
 Majuskeln *siehe* Schriftauszeichnung
 Makro
 → [Index der Befehle etc.](#) 136
 Mathematiksatze 20, 66, 94
 Einheiten *siehe* Einheiten
 Schrift 93
 Trennzeichen *siehe* Trennzeichen
MiKTeX (Distribution) 9–12, 98, 112, 119
MiKTeX Portable (Distribution) 12, 121
 Minimalbeispiel 70–86, 101
 Abschlussarbeit 78
 Aufgabenstellung 80
 Aushang 84
 Dissertation 75
 Dokument 70
 Gutachten 82
 Kollaboratives Schreiben 78, 80
 Minuskeln *siehe* Schriftauszeichnung

N

Nutzerinstallation (lokal) 119–120

O

OpenType-Schriften 21
 Optionen 16
 → [Index der Optionen](#) 135
 Outline-Eintrag *siehe* Lesezeichen

P

Pakete
 → [Index der Dateien etc.](#) 143
 Papierformat 109
 Parameter
 → [Index der Befehle etc.](#) 136
 Posterklasse 55
 Professor *siehe* Hochschullehrer

Q

Quelltextdokumentation 100
 Querbalken 25
 Querverweise 89, 99–100, 105

R		
Randnotizen	100	
Rechtschreibung	92	
Referent	34, 49	
S		
Satzspiegel	38–40, 53, 88	
doppelseitig	26, 30, 35–38	
Schrift	16–21	
Befehle	19	
Fließtext	17, 19	
Griechische Buchstaben	20	
Mathematiksatz	20, 93	
Schalter	19	
Überschriften	18	
Schriftart	93	
Schriftauszeichnung	18, 88, 105	
Schriftelemente	18, 25, 36, 40	
→ Index der Schriftelemente	141	
Schriftgröße	53, 93	
Schriftstärke	17, 19, 93	
Seitenränder	<i>siehe</i> Satzspiegel	
Seitenstil	38	
Selbstständigkeitserklärung	43, 45–47, 51	
Silbentrennung	<i>siehe</i> Worttrennung	
Sonderzeichen	93	
Sperrvermerk	43, 45, 47, 51	
Sprachunterstützung	90	
Lokalisierung	<i>siehe</i> Bezeichner	
Worttrennung	<i>siehe</i> Worttrennung	
Sprungmarken	<i>siehe</i> Lesezeichen	
Symbole	109	
Symbolverzeichnis	95	
T		
Tabellen	95, 97, 106	
Beschriftung	<i>siehe</i> Gleitobjekte	
Tausendertrennzeichen	<i>siehe</i> Trennzeichen	
Teileseiten	23, 36, 37–38	
Teileüberschriften	23	
T_EX Live (Distribution)	9–12, 119	
T_EX Live Portable (Distribution)	12, 121	
T_EXlipse (Anwendungssoftware)	103	
T_EXmaker (Anwendungssoftware)	103	
T_EXnicCenter (Anwendungssoftware)	103	
T_EXstudio (Anwendungssoftware)	103	
T_EXworks (Anwendungssoftware)	103	
Titel	22, 29–36, 38, 48–50, 52, 123	
Felder	32–35	
Kollaboratives Schreiben	32	
Position	23	
Trennzeichen	32	
Umschlagseite ...	<i>siehe</i> Umschlagseite	
Trennmuster	<i>siehe</i> Worttrennung	
Trennzeichen	66, 94	
Dezimaltrennzeichen	66	
Tausendertrennzeichen	66	
Tutorial	86	
Abschlussarbeit	86	
Mathematiksatz	86	
Trennzeichen Mathematikmodus ...	86	
Typisierung	33, 49–50	
Typografie	90–92, 108	
U		
Überschriften	18, 23	
Fußnoten	48	
Position	23	
Umgebungen		
→ Index der Befehle etc.	136	
Umschlagseite	23, 29, 48, 52	
Update	10	
Nutzerinstallation	120	
V		
Vakatseiten	36–38	
Vektorgrafiken	<i>siehe</i> Grafiken	
Version		
→ Änderungsliste	146	
Verzeichnisse	94	
W		
WinEdt (Anwendungssoftware)	103	
Worttrennung	89, 91, 92	
Z		
Zählervmanipulation	99	
Zeichensatzkodierung	90	
Zeilenabstand	91	
Zitate	89, 92	
Zusammenfassung	40–43	
Zweispaltensatz	30, 40, 42, 67, 89, 92	
Zweitlogo	27, 28	

KLASSEN- UND PAKETOPTIONEN

A

`abstract` 40–42, 41, 43–44
`automark` 26, 41, 44

B

`barfont` *siehe* `cdhead`
`BCOR` 16, 39–40

C

`captions` 97
`cd` 17–18, 22, 23–24, 31, 36, 63
`cdchapter` 23
`cdcover` 23, 31
`cdfont` 17–18, 20, 25, 29–31, 65
`cdfoot` 24–25, 26–27, 29–31, 39
`cdgeometry` 31, 38–39, 53
`cdhead` ... 17, 22, 24, 25–26, 27, 29–31, 65
`cdmath` 18, 20, 65, 93
`cdpart` 23
`cdsection` 23
`cdstyle` 55
`cdtitle` 22–23, 30
`chapterpage` 23–24, 36–37, 38
`chapterprefix` 37
`clearcolor` 30, 38
`cleardoublepage` 37
`cleardoublespecialpage` 36, 37–38

D

`ddc` 27, 28–29, 30–31
`ddcfoot` 28–29, 30–31, 53
`ddchead` 27, 28–29, 30–31
`declaration` 43–45, 44, 46–47
`DIV` 38

E

`extrabottommargin` 27, 29, 39

F

`fontsize` 53, 110
`footinclude` 39

`footnotes` 48

G

`geometry` *siehe* `cdgeometry`
`german` 90

H

`headinclude` 39

M

`manualmark` 26, 41

N

`ngerman` 90

O

`oldcolors` 65
`open` 30, 36–39

P

`paper` 109
`parskip` 16, 42, 89, 107
`parttitle` 36

R

`RGB` 65

S

`sansmath` *siehe* `cdmath`
`slantedgreek` 18, 20, 21, 65
`subjectthesis` .. 33, 34, 35, 46–47, 50–51, 60

T

`titlepage` 23–24, 30–31, 42–43, 45
`tudbookmarks` 48, 52, 89
`tudscrver` 52–53
`twocolumn` 30, 42, 45, 89
`twoside` 26, 30, 36–38

W

`widehead` *siehe* `cdhead`

UMGEBUNGEN, BEFEHLE UND SEITENSTILE

A

`\abovecaptionskip` (Länge) 53
`abstract` 40–41, 42–43, 45–46, 89
 `columns` (Parameter) 42–43
 `language` (Parameter) 42–43
 `markboth` (Parameter) 41, 42–43, 46
 `option` (Parameter) 42–43
 `pagestyle` (Parameter) 42–43
`\addchap` 22–23
`\addpart` 22–23
`\addtokomafont` 40, 106
`\advisor` 32, 34, 49
`\afterpage` 101
`\and` 23, 32, 33–34, 46, 49, 58, 60, 78
`\author` 23, 32, 33–35, 46, 57–58, 78
`\authormore` 32
`\autoref` 100, 105

B

`\belowcaptionskip` (Länge) 53
`\bfdefault` 93
`biburllcpenalty` (Zähler) 105
`biburlnumpenalty` (Zähler) 105
`biburlucpenalty` (Zähler) 105
`\bigskip` 53
`\bigskipamount` (Länge) 53
`\blocking` 43–46, 47, 51
 `columns` (Parameter) 47
 `company` (Parameter) 46, 47, 51
 `language` (Parameter) 47
 `markboth` (Parameter) 47
 `option` (Parameter) 47
 `pagestyle` (Parameter) 47
`\bm` 94
`\boldmath` 17, 20
`\boldsymbol` 94
`bottomnumber` (Zähler) 111
`\bottomrule` 96
`\branch` *siehe* `\discipline`

C

`\caption` 53, 97
`\cdfont` 19
`\cefoot` 24
`\cfoot` 24
`\chair` 24, 25

`\chairman` 57, 58, 61
`\chapter` 22–23, 41, 44
`\chapterheadingskip`
 siehe `\pageheadingsvskip`
`\chapterpagestyle` 37
`\cmidrule` 96
`\cofoot` 24
`\columnsep` (Länge) 53
`\commaswap` 67
`\company` 47, 51
`\confirmation` 43–45, 46, 47, 51
 `closing` (Parameter) 46
 `columns` (Parameter) 46
 `language` (Parameter) 46
 `markboth` (Parameter) 46
 `option` (Parameter) 46
 `pagestyle` (Parameter) 46
 `place` (Parameter) 46
 `supporter` (Parameter) 46, 51
`\confirmationclosing` 46, 47
`\contact` *siehe* `\contactperson`
`\contactperson` 59, 60, 62
`\counterwithout*` 107
`\course` 57, 58, 61
`\coverpagebottommargin` 31
`\coverpageleftmargin` 31
`\coverpagerightmargin` 31
`\coverpagetopmargin` 31

D

`\date` 23, 34–35, 46, 50, 59, 89
`\dateofbirth` 32, 50, 78, 89
`\dblfloatpagefraction` 110–111
`dbltopnumber` (Zähler) 111
`\declaration` 43–45, 47
 `closing` (Parameter) 47
 `columns` (Parameter) 47
 `company` (Parameter) 47
 `language` (Parameter) 47
 `markboth` (Parameter) 44, 47
 `option` (Parameter) 47
 `pagestyle` (Parameter) 47
 `place` (Parameter) 47
 `supporter` (Parameter) 47
`declarations` 43, 45–46, 47

- closing (Parameter) 45–46
- columns (Parameter) 45–46, 47
- company (Parameter) 45–46
- language (Parameter) 45–46, 47
- markboth (Parameter) 45–46, 47
- option (Parameter) 45–46, 47
- pagestyle (Parameter) 45–46, 47
- place (Parameter) 45–46
- supporter (Parameter) 45–46
- \dedication 23, 35
- \defaultthyphenchar 104
- \defensedate 34–35, 50, 89
- \degree *siehe* \graduation
- \department 24, 25
- description 95
- \dinbn 19, 66
- \discipline 57, 58, 61
- \dots 92, 108
- \dotswap 67
- \duedate 57, 58, 61, 89

E

- \email *siehe* \emailaddress
- \emailaddress 60
- \emph 105
- empty (Seitenstil) 24
- empty.tudheadings (Seitenstil) 24–25
- \enquote 92
- enumerate 95
- evaluation 32, 58–59, 62, 82
 - grade (Parameter) 58–59, 62
 - headline (Parameter) 58–59, 62
- \evaluationform 59, 62, 82
- \extraheadline 25
- \extrarowheight 95
- \extratitle 23, 35

F

- \faculty 24, 25
- float 97
- \floatpagefraction 110–111
- \floatsetup 97
- \footcontent 24–25, 28, 39
- \footcontent* 28
- \footlogo 24, 27, 29–31, 39
- \footlogoheight (Länge) 27, 29, 39
- \footlogosep 27
- \footnote 35

G

- \grade 59, 62
- \graduation 34, 50

H

- headings (Seitenstil) 24, 43, 46
- \headingsvskip (Länge) 23–24
- \headlogo 24, 27, 28–31
- \hskip 106
- \hyphenation 104

I

- \ifdin 18, 66
- \ifoot 24
- \ifpdf 113
- \includegraphics 27, 60, 88
- \includesvg 112
- \institute 24, 25
- \issuedate 57, 58, 61, 89
- itemize 58, 60, 95

K

- \KOMAOption 9
- \KOMAOptions 9

L

- \lefoot 24
- \lofoot 24
- longtabu 97
- \lowertitleback 23, 30, 35, 38

M

- \makecover 22–23, 29, 31
 - cdfont (Parameter) 31
 - cdfoot (Parameter) 31
 - cdgeometry (Parameter) 31
 - cdhead (Parameter) 31
 - ddc (Parameter) 31
 - ddcfoot (Parameter) 31
 - ddchead (Parameter) 31
 - footlogo (Parameter) 31
 - headlogo (Parameter) 31
 - pagenumber (Parameter) 31
- \MakeTextUppercase 18, 19, 66, 88
- \maketitle 22–23, 29, 30, 31, 35
 - cdfont (Parameter) 30
 - cdfoot (Parameter) 30
 - cdhead (Parameter) 30
 - ddc (Parameter) 30

ddcfoot (Parameter) 30
 ddchead (Parameter) 30
 footlogo (Parameter) 30
 headlogo (Parameter) 30
 pagenumber (Parameter) 30
 \maketitleonecolumn 29, 30–31
 \marginnote 100
 \marginpar 100–101
 \mathswapoff 67
 \mathswapon 67
 \matriculationnumber 32, 50, 57, 78
 \matriculationyear 32, 50, 57, 78
 \mddefault 93
 \medskip 53
 \medskipamount (Länge) 53
 \midrule 96
 \multicolumn 96
 \multirow 96

N

\newcolumnntype 95
 \nextabstract 42–43
 \nextdeclaration 45–46
 \NoCaseChange 18, 19, 66, 88
 \noindent 107
 notice 32, 59, 60, 62, 84
 headline (Parameter) 59, 62
 \noticeform 60, 61, 84

O

\office 60
 \ofoot 24

P

\pageheadingsvskip (Länge) 23–24
 \pagestyle 24–25, 37
 \paragraph 17, 23
 \part 22–23, 36
 \partpagestyle 36
 \PassOptionsToPackage 63, 113
 \pdfbookmark 48
 \phone *siehe* \telephone
 \place 46, 47, 59
 \placeofbirth 32, 50, 78
 plain (Seitenstil) 24
 plain.scrheadings (Seitenstil) 24
 plain.tudheadings (Seitenstil) ... 24–25, 36
 \printdate 32, 35, 58, 89

\professor 34, 49, 57–58
 \providcaptionname 48
 pspicture 98, 113
 \publishers 23, 31, 35

Q

quotation 42
 quoting 42, 89

R

\recalcstypearea 91, 106
 \ref 105
 \referee 32, 34, 49, 59
 \refoot 24
 \renewcaptionname 43, 48
 \restriction *siehe* \blocking
 \rmfamily 93
 \rofoot 24

S

scrheadings (Seitenstil) 24
 \section 23, 41, 44
 \setcdcolors 65
 \setchapterpreamble 37, 43
 \setlist 95
 \setpartpreamble 43
 \setstretch 91, 106
 \sffamily 93
 \sisetup 107
 \smallskip 53
 \smallskipamount (Länge) 53
 \subfiles 99
 \subject 23, 31, 33, 34, 46–47, 49–51, 60,
 62, 78
 \subparagraph 17, 23
 \subsection 23
 \subsubsection 23
 \subtitle 23, 35
 \supervisor 32, 34, 49, 57
 \supporter 46, 47, 51
 symbolheadings (Zähler) 48, 107

T

tabbing 95
 \tabcolsep (Länge) 96, 106
 table 97
 \tableofcontents 107
 tabu 97

SPRACHABHÄNGIGE BEZEICHNER

A

\abstractname 41–43, 51
 \advisorname 49
 \advisorothername 49
 \assessmentname 59, 62
 \authorname 55

B

\bachelorthesisname 33, 49–50
 \blockingname 44, 46, 51
 \blockingtext 47, 51–52
 \branchname *siehe* \disciplinename

C

\chairmanname 57, 61
 \confirmationname 44, 46, 51
 \confirmationtext 46, 51–52
 \contactname
 siehe \contactpersonname 55
 \contactpersonname 55, 62
 \contentname 59, 62
 \coursename 61
 \coverpagename 52

D

\dateofbirthtext 50
 \datetext 50–51
 \defensedatetext 50–51
 \degreertext *siehe* \graduationtext
 \diplomathesisname 33, 49–50
 \disciplinename 61
 \dissertationname 33, 49–50
 \duedatetext 61

E

\evaluationname 58, 62
 \evaluationtext 58, 62

F

\focusname 57, 60, 61

G

\gradetext 59, 62
 \graduationtext 50

I

\internshipname 33, 49–50
 \issuedatetext 61

L

\listingname 52
 \listlistingname 52
 \logname 33, 49–50

M

\masterthesisname 33, 49–50
 \matriculationnumbername 50
 \matriculationyearname 50

N

\namesname 61
 \noticename 59, 62

O

\objectivesname 57, 61

P

\placeofbirthtext 50
 \professorname 49, 57
 \professorothername 49
 \projectpapername 33, 49–50

R

\refereename 49, 59
 \refereeothername 49, 59
 \reportname 33, 49–50
 \researchname 33, 49–50
 \restrictionname .. *siehe* \blockingname
 \restrictiontext .. *siehe* \blockingtext

S

\seminarpapername 33, 49–50
 \studentresearchname 33, 49–50
 \supervisorname 49, 57
 \supervisorothername 49, 57

T

\taskname 57, 59, 60–61, 62
 \tasktext 57, 60–61
 \titlename 61
 \titlepagename 52

SCHRIFTELEMENTE

A

author 30

D

date 30

dedication 30

P

parttitle 36, 40

publishers 30

S

subject 30

subtitle 30

T

thesis 30, 36, 40

title 30

titlehead 30

titlepage 30, 36, 40

tudheadings 24, 25, 28, 40

FARBEN

HKS07	63, 64	HKS44	63, 64
HKS33	63, 64	HKS57	63, 64
HKS36	63, 64	HKS65	63, 64
HKS41	22–23, 28, 63	HKS92	63

KLASSEN, PAKETE UND DATEIEN

A

acro (Paket) 95
acronym (Paket) 95
afterpage (Paket) 101
amsmath (Paket) 20, 94
array (Paket) 95
auto-pst-pdf (Paket) 98, 113–114

B

babel (Paket) 29, 42, 46, 48, 90, 91, 104
balance (Paket) 92
beamer (Klasse) 9
biblatex (Paket) 92, 95, 103, 105, 112
blindtext (Paket) 70
bm (Paket) 94
bookmark (Paket) 89, 99
booktabs (Paket) 96

C

calc (Paket) 99
caption (Paket) 97
cfr-lm (Paket) 93
chngcntr (Paket) 99, 107
cleveref (Paket) 100
cmbright (Paket) 11, 17, 20–21, 88, 121
cm-super (Paket) 11, 88
coaseoul (Paket) 101
csquotes (Paket) 92, 95

D

dehyph-exptl (Paket) 91, 104
DIN_Bd_PS.zip (Datei) 10, 12, 119
DIN_TTF.zip (Datei) 21
dprogress (Paket) 101

E

ellipsis (Paket) 92, 108
enumitem (Paket) 95
environ (Paket) 89, 121
etoolbox (Paket) 88, 121

F

filecontents (Paket) 112
filemod (Paket) 101, 112
fix-cm (Paket) 110
flafter (Paket) 97, 110
float (Paket) 101

floatrow (Paket) 97, 101, 113
fontenc (Paket) 90, 91, 104
fontinst (Paket) 11–12, 88, 119, 121
fontspec (Paket) 10, 18, 21, 93, 126
fontware (Paket) 11, 88

G

geometry (Paket) 39, 88
german (Paket) 90
glossaries (Paket) 10, 89, 95, 112, 128
graphicx (Paket) 27, 88

H

hfbright (Paket) 11, 88
hyperref (Paket) ... 48, 60, 89, 99–101, 105, 110, 112
hyphsubst (Paket) 90, 91, 104

I

icomma (Paket) 66, 94
ifpdf (Paket) 113
inkscape.exe (Datei) 112
inputenc (Paket) 90
ionumbers (Paket) 67, 94
isodate (Paket) 32, 35, 58, 89
iwona (Paket) 11, 17, 20–21, 88, 121

K

keyval (Paket) 9, 88, 100
koma-script (Paket) 88, 121
kvoptions (Paket) 100
kvsetkeys (Paket) 88, 100

L

l2kurz.pdf (Datei) 3
l2tabu.pdf (Datei) 3
libertine (Paket) 93
libgreek (Paket) 93
listings (Paket) 52, 95, 100, 101
lmodern (Paket) 11, 17, 88, 93
longtable (Paket) 96
ltablex (Paket) 96
ltxtable (Paket) 96, 97
lua-check-hyphen (Paket) 92

M

marginnote (Paket) 100, 101

mathastext (Paket) 93
mathswap (Paket) 66–67, 86, 94, 130
mathswap.pdf (Tutorial) 67, 86
mathtools (Paket) 94
mathtype.pdf (Tutorial) 86
microtype (Paket) 90
miktex-portable.cmd (Datei) 121
morewrites (Paket) 113
mparhack (Paket) 100, 101
mptopdf (Paket) 121
multicol (Paket) 29, 43, 46, 89, 92
multirow (Paket) 96
mwe (Paket) 101
mweights (Paket) 93

N

newtx (Paket) 93
newtxmath (Paket) 93
ngerman (Paket) 90
noindentafter (Paket) 92, 107
nomenc1 (Paket) 95, 128

P

pagecolor (Paket) 101
pdfpages (Paket) 101
pdftricks2 (Paket) 98
pgfkeys (Paket) 100
placeins (Paket) 97, 110
polyglossia (Paket) 90
pst-pdf (Paket) 98
pstricks (Paket) 21, 98, 99

Q

quoting (Paket) 42, 89

R

ragged2e (Paket) 89
relsize (Paket) 93

S

sansmath (Paket) 93
sansmathfonts (Paket) 93
scrartcl (Klasse) 13, 40, 88
scrbase (Paket) 88, 100
scrbook (Klasse) 13, 40, 88
scrhack (Paket) 101
scrlayer-scrpage (Paket) 24–26, 39, 88
scr1tr2 (Klasse) 9, 88
scrreprt (Klasse) 13, 40, 88

scrwfile (Paket) 88, 113
selinput (Paket) 90
selnolig (Paket) 92
setspace (Paket) 91, 106
sfmath (Paket) 93
siunitx (Paket) 94, 107
spelling (Paket) 92
standalone (Paket) 98, 99
subcaption (Paket) 97
subfig (Paket) 97
subfiles (Paket) 98, 99
svg (Paket) 112
svg-inkscape (Paket) 112
symbols-a4.pdf (Datei) 109

T

tabu (Paket) 96, 97, 106
tabularborder (Paket) 96, 106
tabularx (Paket) 96, 97, 106
tabulary (Paket) 96, 106
textcase (Paket) 18, 66, 88, 121
textcomp (Paket) 93
tikz (Paket) 65, 98, 99, 114
tl-tray-menu.exe (Datei) 121
todonotes (Paket) 100, 112, 114
treatise.pdf (Tutorial) 13, 86, 90, 98
trimspaces (Paket) 89, 121
tudbeamer (Klasse) 9, 66, 127
tudbook (Klasse) 9, 66, 127, 128
tudfax (Klasse) 9, 66, 127
tudfonts_install.bat (Datei) 66
tudfonts_install.sh (Datei) 66
tudform (Klasse) 9, 66, 127
tudhaus (Klasse) 9, 66, 127
tudletter (Klasse) 9, 66, 127
tudmathposter (Klasse) 9, 66
tudposter (Klasse) 66
tudscr4lyx.zip (Datei) 103
tudscr4texstudio.zip (Datei) 103
tudscrartcl (Klasse) .. 10, 13, 16–53, 36–37, 39–41, 44, 55
tudscrartclold (Klasse) 120
tudscrbase (Paket) 65
tudscrbook (Klasse) 10, 13, 16–53, 30, 36–37, 39–41, 44, 51
tudscrbookold (Klasse) 120
tudscrcolor (Paket) 40, 63–65, 89, 130

tudscrcomp (Paket)	120, 127–128, 130
tudscrfonts (Paket)	65–66, 130
tudscrfonts_install.bat (Datei)	12, 121
tudscrfonts_install.sh (Datei)	12
tudscrfonts.zip (Datei)	12, 119
tudscrposter (Klasse)	55
tudscrreprt (Klasse)	10, 13, 16–53, 40–41, 44
tudscrreprtold (Klasse)	120
tudscrsupervisor (Paket)	22, 32, 48, 50, 57–62, 60, 80, 82, 84, 89, 124–125, 128, 130
tudscr_uninstall.bat (Datei)	118, 120
tudscr_uninstall.sh (Datei)	118, 120
tudscr_v1.0old_install.bat (Datei)	120
tudscr_v1.0old_install.sh (Datei)	120
tudscr_v2.05_install.bat (Datei)	119
tudscr_v2.05_install.sh (Datei)	119
tudscr_v2.05_update.bat (Datei)	120
tudscr_v2.05_update.sh (Datei)	120
tudscr_v2.05.zip (Datei)	119–120
twocolfix (Paket)	67, 130
typearea (Paket)	38–39, 88, 106
U	
units (Paket)	94, 107
Univers_8_TTF.zip (Datei)	21
Univers_PS.zip (Datei)	10, 12, 119
V	
varioref (Paket)	99, 100
W	
widetable (Paket)	96
X	
xcolor (Paket)	63, 65, 89, 121
xkeyval (Paket)	100
xparse (Paket)	100, 113
xpunctuate (Paket)	92, 108
xspace (Paket)	92, 108
Z	
ziffer (Paket)	66–67, 94

ÄNDERUNGSLISTE

v2.00

Allgemein

Änderungen gegenüber der vorhergehenden Version	123
Generalüberholung und komplette Neuimplementierung von TUD-KOMA-Script	1
Robustheit vieler Befehle und Optionen erhöht	16
Schriften – insbesondere für den mathematischen Satz – verbessert	16

Implementierung

<code>abstractdouble</code> (Option): <i>entfällt</i>	<i>siehe</i> <code>abstract</code> (Option)
<code>abstracttotoc</code> (Option): <i>entfällt</i>	<i>siehe</i> <code>abstract</code> (Option)
<code>\authortext</code> (Bezeichner): <i>entfällt</i>	123
<code>\birthday</code> (Befehl): <i>entfällt</i>	<i>siehe</i> <code>\dateofbirth</code> (Befehl)
<code>\birthdaytext</code> (Bezeichner): <i>entfällt</i>	<i>siehe</i> <code>\dateofbirthtext</code> (Bezeichner)
<code>\birthplace</code> (Befehl): <i>entfällt</i>	<i>siehe</i> <code>\placeofbirth</code> (Befehl)
<code>\birthplacetext</code> (Bezeichner): <i>entfällt</i>	<i>siehe</i> <code>\placeofbirthtext</code> (Bezeichner)
<code>bookmarks</code> (Option): <i>entfällt</i>	<i>siehe</i> <code>tudbookmarks</code> (Option)
<code>cd</code> (Option): <i>entfällt</i>	123
<code>cdfonts</code> (Option): <i>entfällt</i>	<i>siehe</i> <code>cdfont</code> (Option)
<code>cdtask</code> (Option): <i>entfällt</i>	<i>siehe</i> <code>task</code>
<code>cdtitle</code> (Option): <i>entfällt</i>	123
<code>chapterclear</code> (Option): <i>entfällt</i>	<i>siehe</i> <code>cleardoublespecialpage</code> (Option)
<code>color</code> (Option): <i>entfällt</i>	<i>siehe</i> <code>cd</code> (Option)
<code>\confirmationandrestriction</code> (Befehl): <i>entfällt</i>	<i>siehe</i> <code>\declaration</code> (Befehl)
<code>\defensetext</code> (Bezeichner): <i>entfällt</i>	<i>siehe</i> <code>\defensedatetext</code> (Bezeichner)
<code>\duetext</code> (Bezeichner): <i>entfällt</i>	<i>siehe</i> <code>\duedatetext</code> (Bezeichner)
<code>\enrolmentname</code> (Bezeichner): <i>entfällt</i>	<i>siehe</i> <code>\matriculationyearname</code> (Bezeichner)
<code>\enrolmentyear</code> (Befehl): <i>entfällt</i>	<i>siehe</i> <code>\matriculationyear</code> (Befehl)
<code>\focustext</code> (Bezeichner): <i>entfällt</i>	<i>siehe</i> <code>\focusname</code> (Bezeichner)
<code>headfoot</code> (Option): <i>entfällt</i>	123
<code>\location</code> (Befehl): <i>entfällt</i>	<i>siehe</i> <code>\place</code> (Befehl)
<code>\logfile</code> (Befehl): <i>entfällt</i>	<i>siehe</i> <code>\headlogo</code> (Befehl)
<code>\objectivestext</code> (Bezeichner): <i>entfällt</i>	<i>siehe</i> <code>\objectivesname</code> (Bezeichner)
<code>partclear</code> (Option): <i>entfällt</i>	<i>siehe</i> <code>cleardoublespecialpage</code> (Option)
<code>\restrictionandconfirmation</code> (Befehl): <i>entfällt</i>	<i>siehe</i> <code>\declaration</code> (Befehl)
<code>\signatureheight</code> (Länge): <i>entfällt</i>	124
<code>\startdate</code> (Befehl): <i>entfällt</i>	<i>siehe</i> <code>\issuedate</code> (Befehl)
<code>\starttext</code> (Bezeichner): <i>entfällt</i>	<i>siehe</i> <code>\issuedatetext</code> (Bezeichner)
<code>\studentid</code> (Befehl): <i>entfällt</i>	<i>siehe</i> <code>\matriculationnumber</code> (Befehl)
<code>\studentidname</code> (Bezeichner): <i>entfällt</i>	<i>siehe</i> <code>\matriculationnumbername</code> (Bezeichner)
<code>\submissiondate</code> (Befehl): <i>entfällt</i>	<i>siehe</i> <code>\date</code> (Befehl)
<code>\submissiontext</code> (Bezeichner): <i>entfällt</i>	<i>siehe</i> <code>\datetext</code> (Bezeichner)
<code>\supervisorIIname</code> (Befehl): <i>entfällt</i>	<i>siehe</i> <code>\supervisorothername</code> (Bezeichner)
<code>\taskcolwidth</code> (Länge): <i>entfällt</i>	124
<code>taskcompact</code> (Option): <i>entfällt</i>	124
<code>\tasks</code> (Befehl): <i>entfällt</i>	<i>siehe</i> <code>\taskform</code> (Befehl)
<code>\titlecoldelim</code> (Bezeichner): <i>entfällt</i>	<i>siehe</i> <code>\titledelimiter</code> (Befehl)

\titlecolwidth (Länge): <i>entfällt</i>	123
tudfonts (Option): <i>entfällt</i>	siehe cdfont (Option)
tudfoot (Option): <i>entfällt</i>	siehe cdfoot (Option)
v2.01	
Allgemein	
TUD-KOMA-Script-Bundle auf CTAN veröffentlicht	10
Unterschneidung (Kerning) der Ziffern für Univers verbessert	16
Implementierung	
\maketitle (Befehl): Bugfix für Schriftstärke auf Titelseite	30
\subtitle (Befehl): Bugfix für Schriftstärke bei Verwendung des Untertitels	35
v2.02	
Allgemein	
Änderungen gegenüber der vorhergehenden Version	125
fontspec (Paket): Beschreibung hinzugefügt	93
Funktionalität im Dokument umschaltbar	67
Installationsroutine der PostScript-Schriften angepasst	10
listings (Paket): Beschreibung ergänzt	100
mwe (Paket): Beschreibung hinzugefügt	101
noindentafter (Paket): Beschreibung hinzugefügt	92
OpenType-Schriften mit fontspec (Paket) verwendbar	21
scrlayer-scrpage (Paket): Für Verwendung von TUD-KOMA-Script notwendig	88
standalone (Klasse): Probleme behoben	99
subfiles (Paket): Beschreibung hinzugefügt	99
tabu (Paket): Verwendung nur bedingt empfehlenswert	97
Tipps & Tricks: Bezeichnungen der Gliederungsebenen	105
Tipps & Tricks: Das Setzen von Auslassungspunkten	108
Tipps & Tricks: Fehler beim Schreiben von Hilfsdateien	112
Tipps & Tricks: Hinweis auf Rand bei mehrzeiligen Einträgen ergänzt	107
Tipps & Tricks: Hinweise zu auto-pst-pdf (Paket)	113
Tipps & Tricks: Literaturverwaltung	103
Tipps & Tricks: Lokale Änderungen	105
Tipps & Tricks: URL-Umbrüche im Literaturverzeichnis	105
Tipps & Tricks: Warnung beim Erzeugen des Inhaltsverzeichnisses	107
Tipps & Tricks: Worttrennungen	104
Umbenennung einiger Befehle für Kompatibilität mit anderen Paketen	16
xkeyval (Paket): Beschreibung hinzugefügt	100
Implementierung	
abstract (Umgebung): Trennung einzelner Teile mit \nextabstract (Befehl)	42
abstract (Option): Wert double mit multi ersetzt	40
abstract (Option): Wert markboth neu	40
abstract (Option): Wert tocleveledown neu	40
abstract (Umgebung): markboth (Parameter): neu	42
abstract (Umgebung): pagestyle (Parameter): neu	42
\blocking (Befehl): neu	47
\blocking (Befehl): columns (Parameter): neu	47
\blocking (Befehl): markboth (Parameter): neu	47
\blocking (Befehl): pagestyle (Parameter): neu	47

<code>\maketitle</code> (Befehl): <code>pagenumber</code> (Parameter): <i>neu</i>	30
<code>\mathswapoff</code> (Befehl): <i>neu</i>	67
<code>\mathswapon</code> (Befehl): <i>neu</i>	67
<code>\nextabstract</code> (Befehl): <i>neu</i>	42
<code>\pageheadingsvskip</code> (Länge): <i>neu</i>	23
<code>\partpagestyle</code> (Befehl): <code>plain.tudheadings</code> (Seitenstil) wird genutzt	36
<code>parttitle</code> (Schriftelement): <i>neu</i>	36
<code>\phone</code> (Befehl): <i>entfällt</i>	<i>siehe \telephone</i> (Befehl)
<code>plain.tudheadings</code> (Seitenstil): <i>neu</i>	24
<code>\professorname</code> (Bezeichner): Unterscheidung von einem und mehreren Professoren ..	49
<code>\professorothername</code> (Bezeichner): <i>neu</i>	49
<code>\refereename</code> (Bezeichner): Unterscheidung von einem und mehreren Gutachtern	49
<code>\restriction</code> (Befehl): <i>entfällt</i>	<i>siehe \blocking</i> (Befehl)
<code>\restrictionname</code> (Bezeichner): <i>entfällt</i>	<i>siehe \blockingname</i> (Bezeichner)
<code>\restrictiontext</code> (Bezeichner): <i>entfällt</i>	<i>siehe \blockingtext</i> (Bezeichner)
<code>symbolheadings</code> (Zähler): <i>neu</i>	48
<code>\telephone</code> (Befehl): <i>neu</i>	60
<code>thesis</code> (Schriftelement): <i>neu</i>	36
<code>titlepage</code> (Schriftelement): <i>neu</i>	36
<code>tudheadings</code> (Seitenstil): <i>neu</i>	24
<code>tudpage</code> (Umgebung): <code>foot</code> (Parameter) <i>entfällt</i>	29
<code>tudpage</code> (Umgebung): <code>head</code> (Parameter) <i>entfällt</i>	29
<code>tudpage</code> (Umgebung): <code>ddc</code> (Parameter): <i>neu</i>	29
<code>tudpage</code> (Umgebung): <code>ddcfoot</code> (Parameter): <i>neu</i>	29
<code>tudpage</code> (Umgebung): <code>ddchead</code> (Parameter): <i>neu</i>	29
<code>tudpage</code> (Umgebung): <code>pagestyle</code> (Parameter): <i>neu</i>	29
<code>tudscrfonts</code> (Paket): <i>neu</i>	65

v2.03

Allgemein

Änderungen gegenüber der vorhergehenden Version	126
Anpassungen interner Befehle an KOMA-Script-Version v3.15	16
Bugfix für Umschlagseite und Titel beim Satzspiegel	29
Bugfix für Zusammenfassung, Selbstständigkeitserklärung und Sperrvermerk bei Seitenstil und Kolumnentitel	40
Satzspiegel im Corporate Design geändert, DRESDEN-concept-Logo im Fußbereich wird ohne vergrößerten Seitenrand verwendet	53
Seitenstile um zweifarbigen Kopf und farbigen Fuß erweitert	24

Implementierung

<code>barfont</code> (Option): <i>entfällt</i>	<i>siehe cdhead</i> (Option)
<code>cd</code> (Option): neue Farbvarianten (<code>bicolor</code> und <code>fullcolor</code>)	22
<code>cdchapter</code> (Option): neue Farbvarianten (<code>bicolor</code> und <code>fullcolor</code>)	23
<code>cdcover</code> (Option): neue Farbvarianten (<code>bicolor</code> und <code>fullcolor</code>)	23
<code>cdfoot</code> (Option): farbiger Hintergrund der Fußzeile möglich	26
<code>cdgeometry</code> (Option): <i>neu</i>	38
<code>cdhead</code> (Option): <i>neu</i>	25
<code>cdmath</code> (Option): <i>neu</i>	20
<code>cdpart</code> (Option): neue Farbvarianten (<code>bicolor</code> und <code>fullcolor</code>)	23

cdtitle (Option): neue Farbvarianten (bicolor und fullcolor)	22
extrabottommargin (Option): <i>neu</i>	39
\footlogo (Befehl): <i>neu</i>	27
\footlogoheight (Länge): <i>neu</i>	27
\footlogosep (Befehl): <i>neu</i>	27
geometry (Option): <i>entfällt</i>	<i>siehe cedgeometry (Option)</i>
\makecover (Befehl): cdfoot (Parameter): <i>neu</i>	31
\makecover (Befehl): cdhead (Parameter): <i>neu</i>	31
\makecover (Befehl): ddc (Parameter): <i>neu</i>	31
\makecover (Befehl): ddcfoot (Parameter): <i>neu</i>	31
\makecover (Befehl): ddhead (Parameter): <i>neu</i>	31
\makecover (Befehl): footlogo (Parameter): <i>neu</i>	31
\makecover (Befehl): headlogo (Parameter): <i>neu</i>	31
\maketitle (Befehl): cdfoot (Parameter): <i>neu</i>	30
\maketitle (Befehl): cdhead (Parameter): <i>neu</i>	30
\maketitle (Befehl): ddc (Parameter): <i>neu</i>	30
\maketitle (Befehl): ddcfoot (Parameter): <i>neu</i>	30
\maketitle (Befehl): ddhead (Parameter): <i>neu</i>	30
\maketitle (Befehl): footlogo (Parameter): <i>neu</i>	30
\maketitle (Befehl): headlogo (Parameter): <i>neu</i>	30
sansmath (Option): <i>entfällt</i>	<i>siehe cdmath (Option)</i>
tudpage (Umgebung): color (Parameter) <i>entfällt</i>	29
tudpage (Umgebung): cdfoot (Parameter): <i>neu</i>	29
tudpage (Umgebung): cdhead (Parameter): <i>neu</i>	29
tudpage (Umgebung): footlogo (Parameter): <i>neu</i>	29
tudscrver (Option): <i>neu</i>	52
widehead (Option): <i>entfällt</i>	<i>siehe cdhead (Option)</i>

v2.04

Allgemein

Änderungen gegenüber der vorhergehenden Version	126
Einfachere Verwendung von fontspec (Paket)	18
Installationsskripte robuster und verbessert sowie für die beiden portablen Distributionen T_EX Live Portable und MiK_TE_X Portable erweitert	10
Tipps & Tricks: Vermeiden des Skalierens einer PDF-Datei beim Druck	109
Tipps & Tricks: Warnung bei der Schriftgrößenwahl	110
Unterstützung von tudposter (Klasse) und tudmathposter (Klasse)	66
Werte bestimmter Längen abhängig von der verwendeten Schriftgröße	53

Implementierung

abstract (Option): Wert tocmultiple <i>neu</i>	40
cd (Option): farbiger Querbalken möglich (barcolor)	22
cdchapter (Option): farbiger Querbalken möglich (barcolor)	23
cdcover (Option): farbiger Querbalken möglich (barcolor)	23
\cdfont (Befehl): <i>neu</i>	19
cdhead (Option): farbiger Querbalken möglich (barcolor)	25
cdpart (Option): farbiger Querbalken möglich (barcolor)	23
cdtitle (Option): farbiger Querbalken möglich (barcolor)	22
declaration (Option): Wert tocmultiple <i>neu</i>	43

fontspec (Option): <i>entfällt</i>	126
\footcontent* (Befehl) : <i>neu</i>	28
\footcontent (Befehl): <i>neu</i>	28
\namesname (Bezeichner): <i>neu</i>	61
\nextdeclaration (Befehl): <i>neu</i>	45
\textcdfont (Befehl): <i>neu</i>	19
\TUDClassName (Befehl): <i>neu</i>	130
tudheadings (Schriftelement): <i>neu</i>	25
\TUDScript (Befehl): <i>neu</i>	130
\TUDVersion (Befehl): <i>neu</i>	130
v2.05	
Allgemein	
Einstellungen für den Satzspiegel für die jeweilige ISO/DIN-Klasse des verwendeten Papierformates identisch	53
Tipps & Tricks: Fehler beim Laden eines Paketes mit Optionen	113
Implementierung	
\authorname (Bezeichner): <i>neu</i>	55
cdsection (Option): <i>neu</i>	23
cdstyle (Option): <i>neu</i>	55
\contactname (Bezeichner): <i>neu</i>	55
tudscrposter (Klasse): <i>neu</i>	55
\webpage (Befehl): <i>neu</i>	55